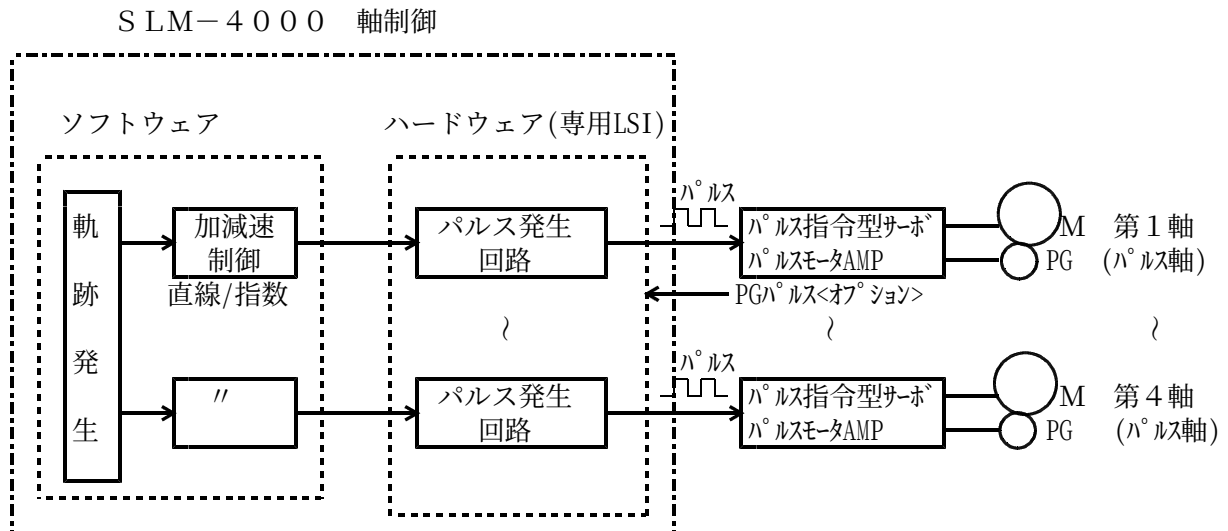


4. サーボ制御（一般仕様）

パルス指令軸制御については、パルス指令形サーボとパルスモータでは、ほとんど同一です。
このマニュアルでは、パルスモータも含めてサーボ制御と呼びます。

サーボ(軸)制御のブロック図



パルス指令軸制御（第1～第4軸）

パルス信号によりサーボ制御をおこないます。（サーボアンプ／パルスモータ）

パルス量 = 移動量

パルス周期 = 移動速度

4-1 座標系

S L Mでは、基本的に2種類の座標系を管理しています。

【機械座標系】

原点復帰位置を原点とします。

ソフトリミットチェックなどはこの座標系でおこないます。

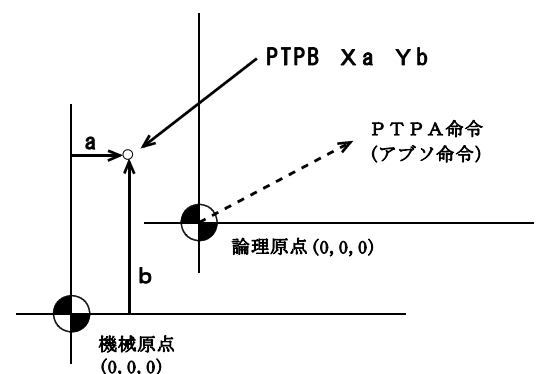
PTPB命令は、この座標系でのアブソ指令です。

【論理座標系】

原点設定（入力又は操作）をした位置を、原点とします。

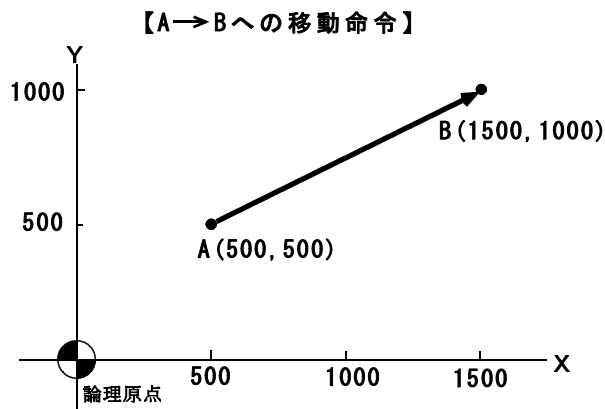
PTPB／LINBを除く全ての移動命令、運転プログラムは、論理座標系での命令です。

原点復帰完了時に、「論理原点＝機械原点」とする方式と論理原点を設定しない方式を選択できます。



4-2 アブソ指令とインクレ指令

移動命令の目標位置の指令方法には、「アブソ指令」と「インクレ指令」があります。



■アブソ指令の表記

PTPA X1500 Y1000
目標位置を指令します。

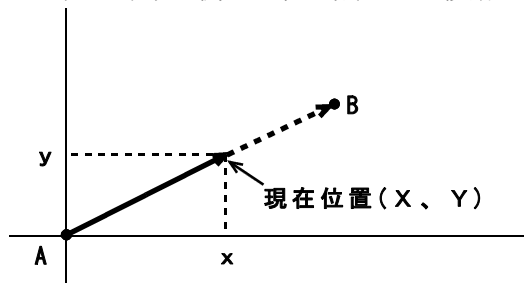
■インクレ指令の表記

PTP X1000 Y500
移動量を指令します。

■相対移動量

命令1ステップ毎の移動量です。

上記の命令の例では、A点からの移動量となります。



(X、Y)は、論理原点からの距離
(論理座標)

(x、y)は、A点からの距離
(相対移動量)

◆1.6

4-2-1 アブソ/インクレで紛らわしい言葉

座標系	座標原点	説明	応用
機械座標系	機械原点 (原点復帰)	機械固有の座標。原点復帰して座標系が決まる。 機能編4-1参照	ホーム位置 ツール交換 ワーク搬入/出 PTPB命令
論理座標系	論理原点	原点設定した位置 G92/CSET命令でその都度変わる 機能編4-1参照	一般的に使う
ワーク座標系	ワーク原点	現状では、ワーク座標系＝論理座標系です。将来G5□命令に対応したときに、論理座標系とは、別のテンポラリな論理座標が生じます。作業やワーク毎に決めます。	図面などはワーク基準で表記。これに合わせやすい。

位置表示名	位置の原点	説明	補足説明
アブソ位置	機械原点 (機械座標系)	機械座標系での位置 ソフトリミットチェックなど	SLMで自動記憶するのが疑似アブソです。
機械位置	論理原点 (論理座標系)	F Bパルスを積算した座標 論理座標系 F B無しの軸では、指令位置をコピー。	指令座標とは、時間ズレあり（サーボのたまり）
指令位置	論理原点 (論理座標系)	サーボへ指令した座標 論理座標系	
偏差		指令位置－機械位置	

注意 **機械位置は、論理座標系です。（機械座標系ではありません）**
オープンMCの習慣でそのようになってます。紛らわしくてすみません。

移動命令名	座標系	説明	詳細説明
アブソ指令 (G90) (テクノ言語では、命令が別)	論理座標系 機械座標系	移動指令の目標を座標で指示 一般は、論理座標 PTPA, LINA, PTMA, G90では論理座標。 PTPB, LINB, G28では、機械座標。	機能編 4－2
インクレ指令 (G91)	(座標は、無関係)	移動指令をそのステップの移動量で指示。PTP、LIN、G91	機能編 4－2 相対移動量

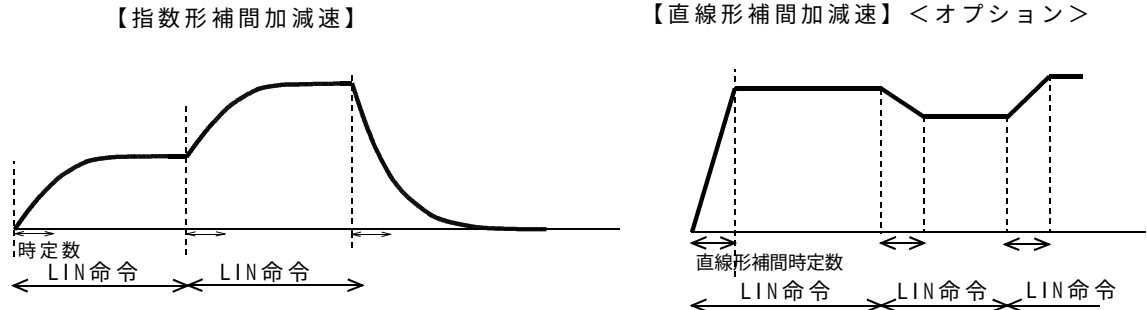
位置検出器	別名	機能	メリット・応用・注意点
アブソエンコーダ	アブソPG 絶対値エンコーダ	電源を落としても座標を忘れない位置検出器です。	原点復帰なしでも機械座標が成立。 安川電機アブソPG対応。
インクレエンコーダ	インクレPG (通常)	電源を落とすと座標が判らなくなります。	機械座標を成立させるには原点復帰が必要。
疑似アブソ	(SLMがバッテリバックアップする情報の一つ)	SLMが電源断のときに記憶しておく座標。もし、電源断中に軸が動かなければ、次の電源投入時に記憶がほぼただしので、機械座標としてとりあえず使う。	電源断では、サーボロックしないので、位置がずれる。また、外力で動かされれば、大きく変化してしまう。

位置検出器のことで、上記の座標系や位置情報名称とは、あまり関係がありません。
もちろん、アブソエンコーダのシステムでは、機械座標系のアブソ位置と、アブソエンコーダの読み出しデータとは、一定のオフセットの関係があります。

4-3 補間加減速 (指数形/直線形/S字)

補間指令(直線補間や円弧補間)に対しては、各軸に加減速制御(スムージング)がかかります。
パラメタとしては、補間時定数を指定できます。
スムージングが不要の場合、補間時定数を0として下さい。

軸速度の様子

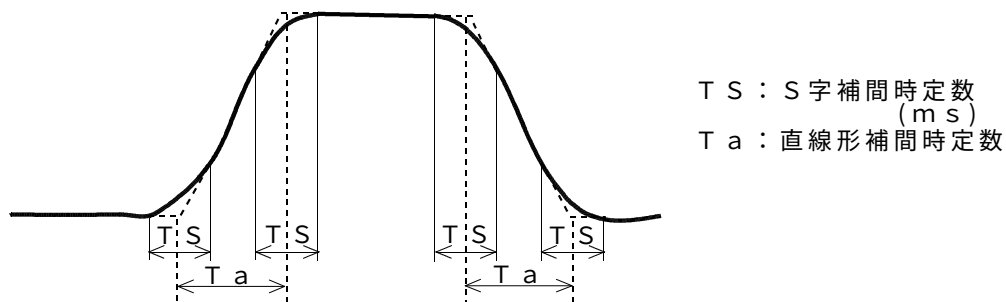


◆1.2

【S字加減速】 <オプション>

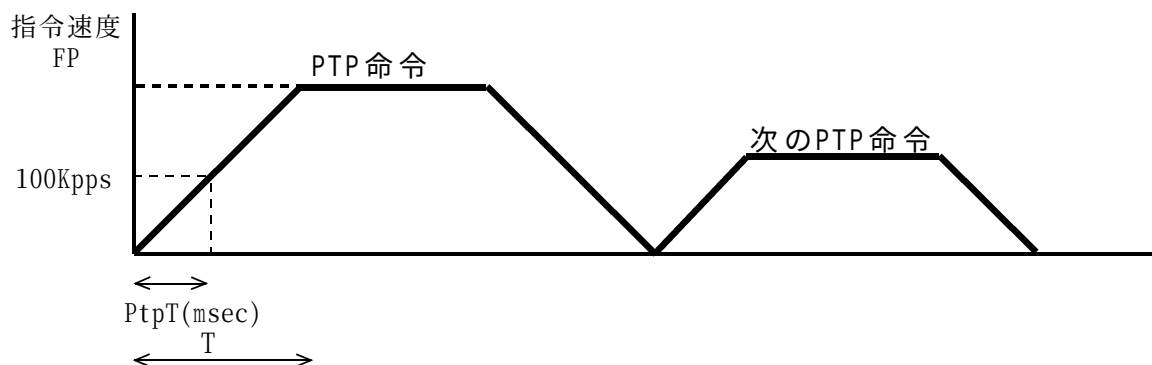
補間加減速(指数形/直線形)をさらにスムージングします。一般には、直線形補間加減速と組み合わせます。

速度の変化点で T_s (ms) の平均化(フィルタリング)をします。



4-4 直線形加減速

PTPの移動指令の際には、直線形加減速制御をおこないます。
制御は各軸独立におこないます。



T : 加減速時間 . . . 0からPTP速度に到達するまでの時間
 $PtpT$: PTP加減速時定数 . . . 0から100Kppsに到達するまでの時間

※TB00-800G-6「8-2送り速度、加減速度などのパラメタ」を参照ください。

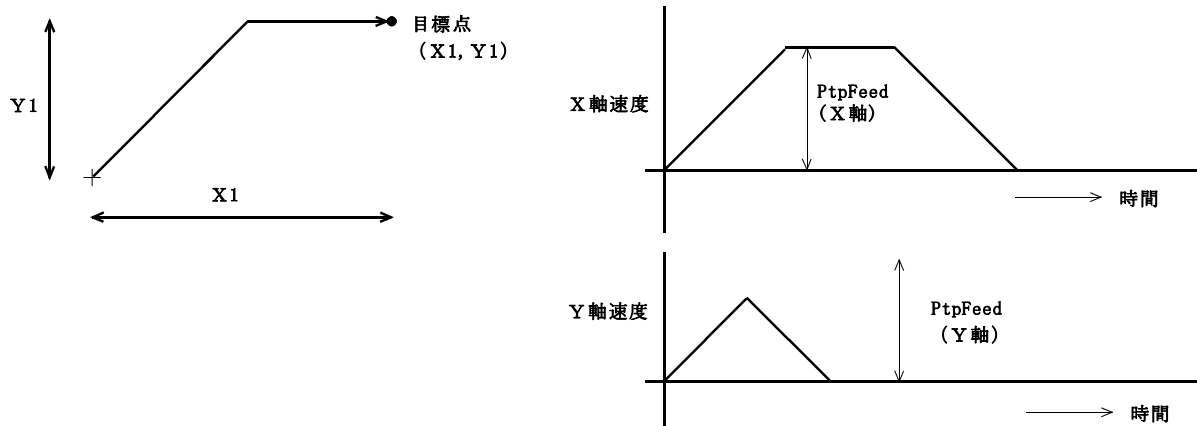
◆2.1

◆1.6

4-5 PTP制御(位置決め) PTP□/G00

目標位置へ向かって、各軸独立に直線加減速制御により位置決めします。
各軸独立ですので、各軸の目標到達時刻には、差が出ます。
PTP速度(各軸)は、サーボパラメタです。

PTP X1,Y1 の例



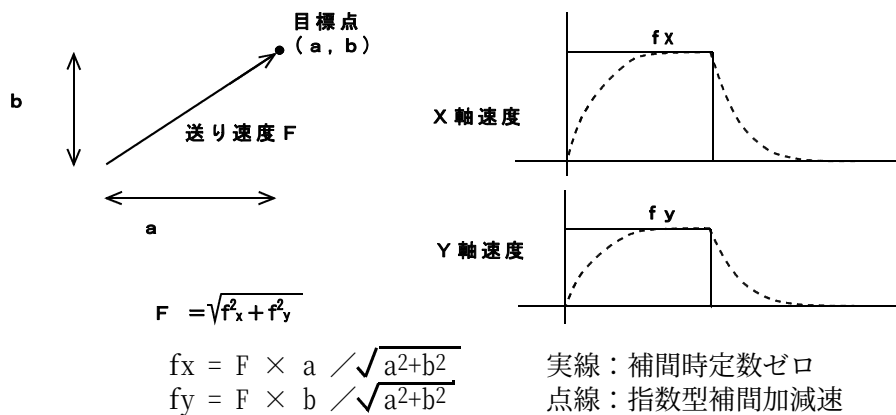
◆1.6

4-6 直線補間 LIN□/G01

1軸～4軸同時の直線補間です。
この場合、目標点到達は、全軸が同時です。
速度指令は、1軸～4軸の合成された接線速度です。
例：4軸の合成速度

$$F = \sqrt{f_x^2 + f_y^2 + f_z^2 + f_a^2} \quad f□: \text{各軸速度}$$

LIN Xa Yb F□: の例



4-7 円弧補間

2軸の円弧補間です。
目標位置と中心座標値を設定します。
速度指令は、2軸の合成された接線速度です。
以下の2つの方式があります。セッティングPCの「インターフェース設定画面」の「円弧処理」で選択下さい。

4-7-1 プリ解析方式

プリ解析方式では、どのような軌跡指令でも完全パス動作します。ただし、展開により実ステップ長が増えます。(5-4プリ開処理参照)
1周円以上の円弧動作をさせる場合は、このコマンドを連続的に使用することで実現できます。ただし、この場合は円弧の切り換わり時に速度が変動する場合があります。御注意下さい。
この方式では、アブソ指令(CIRRA/CIRLA)は、使えません。
速度Fは、ヘリカル軸も含めた接線速度です。

4－7－2 ボード内部方式

ボード内部で円弧解析を行う機能です。

ボード内円弧では、円弧ステップの初めと終了で、微少な速度変動が生じる場合があります。
(通常の使用状態では、問題になりません。)

速度 F は、円弧 2 軸の合成速度です。ヘリカル軸はこれに同期します。

4-8 手動送り

◆1.6

4-8-1 ジョグ(JOG)送り

指定された軸を、ジョグ速度（パラメタ）で指定された方向へ移動します。
軸キー操作、機械パネル I F、コマンド（P C / 通信）で指定します。

Jog Feed ジョグ（手動送り）速度（P/sec）
JOG送り速度

4-8-2 インチング送り（手動）

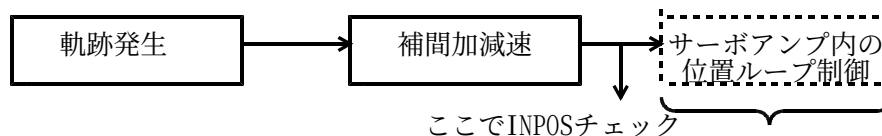
指定された軸を一定のパルス量だけ送ります。

倍 率	送 り 量
× 1	1 パルス送り
× 1 0	1 0 パルス送り
× 1 0 0	1 0 0 パルス送り
× 1 0 0 0	1 0 0 0 パルス送り

4-9 インポジションチェック

<パルス指令軸の場合>

1 ステップ毎の目標位置に対して、軌跡発生パルス払い出し量が、I N P O S 量以内にはいると位置決め完了出力をONします。



この分は、早めにINPOSとなります

<アナログ指令軸の場合>（将来）

目標値と現在値が、インポジションゾーン（パラメタ）に入っているかをチェックします。
偏差レジスタ ≤ I N P O S の時に、I N P O S (出力/ステータス)ONとなります。

<インポジションチェック有/無>

インポジションチェックONの状態では補間命令(L I N / L I R)以外の命令の時は、位置決め完了(I N P O S)を待ってから、次ステップを実行します。
インポジションチェックOFFでは、F Gの軌跡発生終了後、ただちに次ステップを実行します。

インボズ動作比較表

	I N P O S 無効 I N P D / G 6 4	I N P O S 有効(デフォルト) I N P E / G 6 1
P T P → P T P P T P → L I N L I N → P T P	I N P O S チェック無し (擬似つなぎと同様)	I N P O S チェック有り
L I N → L I N L I N → プリ解円弧 プリ解円弧 → L I N	つなぎ	
L I N → 内部円弧 内部円弧 → L I N	擬似つなぎ	

※ 1

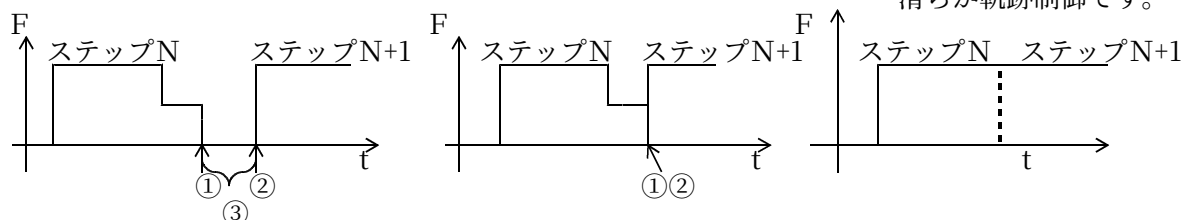
※ 1

※ 1 の動作でINPOSチェック(停止)させたい場合は、補間命令の間にドウェル(G04又はTIM)を入れてください。

実動作説明 ①軌跡発生完了、②次命令開始、③INPOS待ち

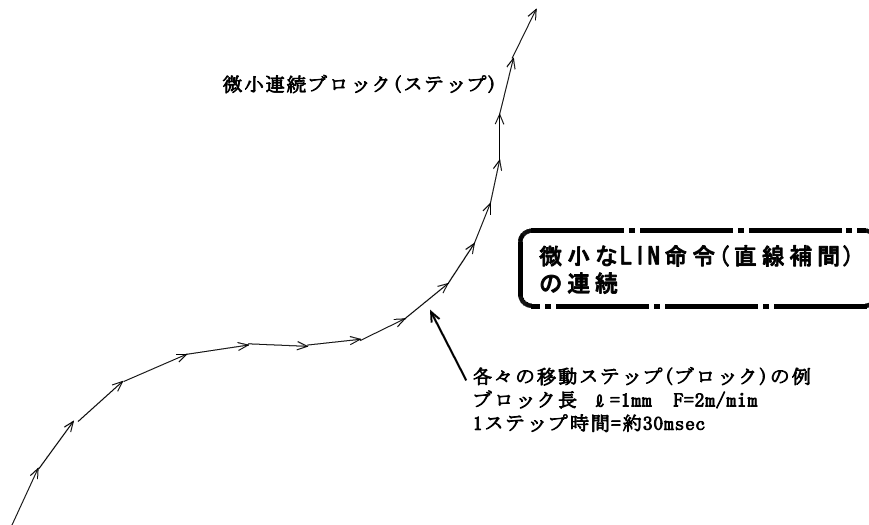
I N P O S チェック有り I N P O S チェック無し(擬似つなぎ)
軌跡発生完了後INPOS待ち 軌跡発生完了と次命令開始が同時
INPOS状態のあと次命令発行。
位置決めに使います。

つなぎ
速度変動が起きないように補正します。
切削や研磨のための滑らか軌跡制御です。



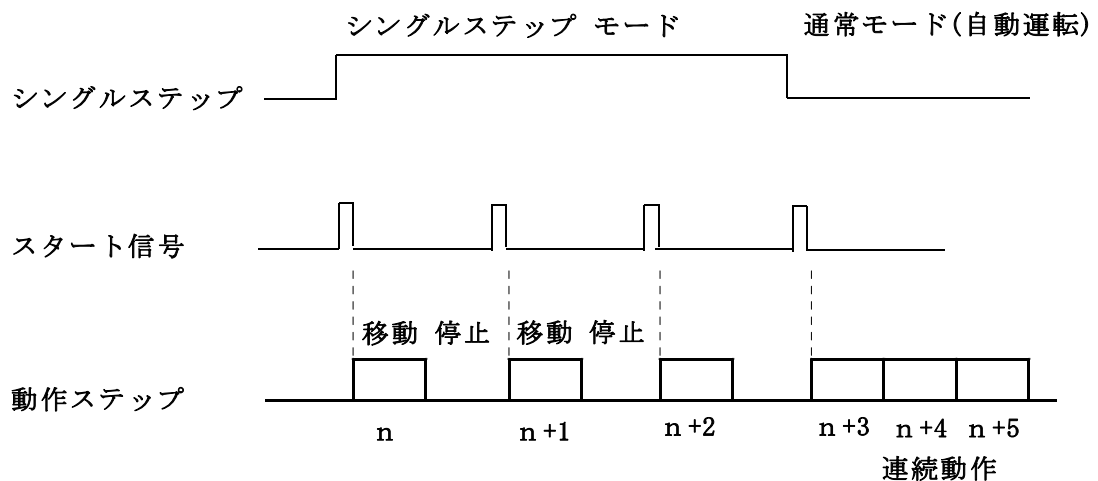
4-10 なめらかな送り制御（パス機能）

各ステップ(ブロック)の実行の間で、停止時間が有りません。
NCのブロック間停止ゼロ（パス機能）に当たります。
このパス機能により微小連続送りでのなめらかな輪郭制御が実現できます。
以下のような軌跡を、指定された接線速度でなめらかに送ります。



4-11 シングルステップ動作

シングルステップモードでは運転プログラムの各ステップ毎に停止し、スタート信号により1ステップずつ進行します。



運転プログラムのセッティングやメンテナンス時に、動作確認のために使用します。

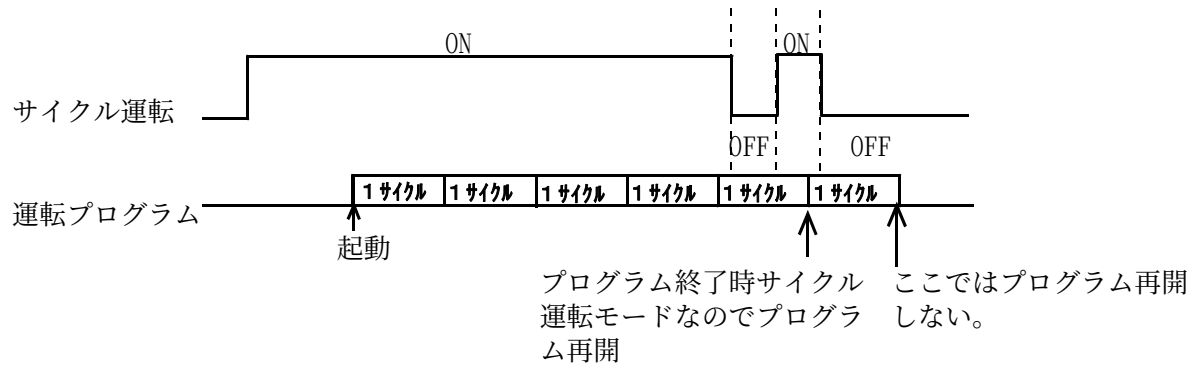
4-1-2 サイクル運転

サイクル運転モードでは運転プログラムの繰り返し実行を行います。

S L Mコントローラは、運転プログラムのE N Dステップ実行時、サイクル運転モードの場合自動的に運転プログラムの先頭から再実行を行います。

繰り返し実行中に、サイクル運転モードをO F Fにすると、その回の最後まで実行した後、プログラム実行終了します。

サイクル運転モードのO N / O F Fは、通信コマンドでおこないます。



サイクル運転モードで、繰り返し実行する運転プログラムの先頭と最後のステップが補間移動の場合でも、最後のステップと先頭ステップのつなぎ処理は行いません。

E N Dステップでは、1 制御周期とインポーズチェック(有効の場合)の時間は、停止します。

4-1-3 オーバーライド機能

送り速度を 1 %～200%の比率で、動作中に変更できます。

この機能もセッティングやメンテナンスで使用します。

(標準操作パネルの S Wでは、25%刻みでの変更)

◆1.6

4-14 ストロークリミット

機械の動作範囲外への移動を禁止するために、二重の安全対策を実施しています。

ソフトウェアストロークリミット (ソフトOT)

ソフトウェアにより、動作可能な範囲を規定。
この領域を越える移動指令があった場合、
SLMは停止して、アラームとなります。
この場合、サーボ主電源は落ちません。そのまま
手動操作(JOG)にて戻せます。

ハードウェアストロークリミット (ハードOT)

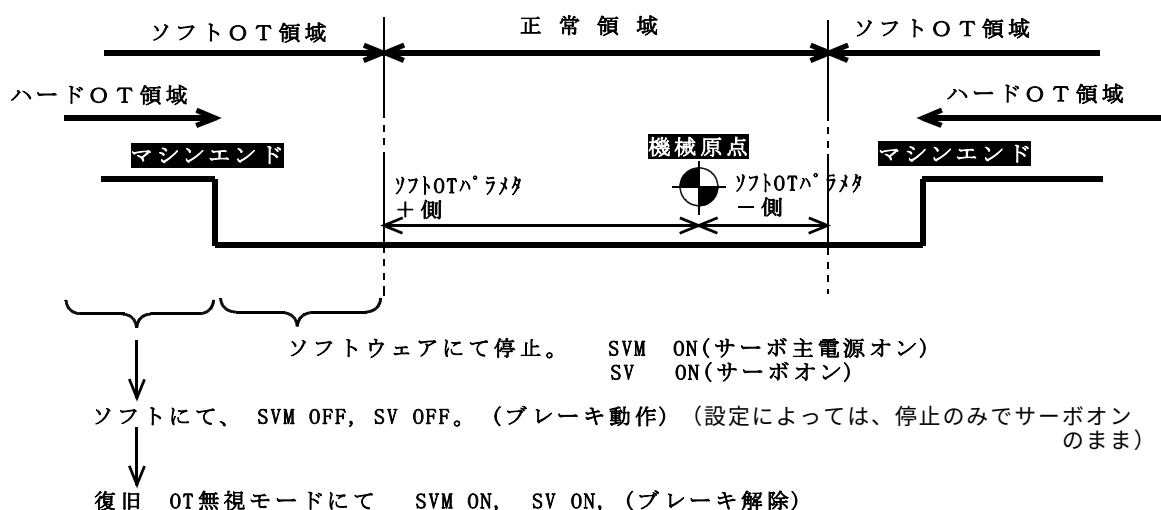
マシンエンドに配置したLS信号をSLMへ入力
します。一般はb接論理で入力し、これがOFF
するとSLMは緊急停止して、サーボ主電源を落
とします。
通常の使用状態では必ずソフトOTにより停止
しますので、ハードOTが作動する可能性はほ
んどありません。

ハードOTの処理

基本ROMSWの選択で以下の方式あります。

OT時サーボOFF	動作	復旧
する	即停止後にサーボ オフ	OT無視モードにして、リセット(サーボオン) ジョグでもどして、リセット。(アラーム解除)
しない	即停止のみ	そのままジョグでもどして、リセット(アラーム 解除)

詳細は、ROMSWマニュアル TB00-0801 を参照ください。



ソフトOTのパラメタは、機械原点を基準に、+側と-側の各々に対して設定します。

注記1

SLMシリーズの標準的使用方法としては、OT入力はSLMにのみ入力して、サーボ
アンプへは入力しません。これは「OT無視モード」にて復旧を可能とするためです。
この方式の問題点としては、SLMのCPUが何かの原因で暴走した場合においては、
OTで止まらないと言う点です。

万が一の安全対策としては以下の方法があります。

- (1) さらに外側に、真のハードOTLSを配置して、リレー回路にてサーボ主電源を落とす。
- (2) ハードOTをサーボアンプへ入力する。

注記2

マシンエンドにおける、メカニカルストッパーは必ず設置して下さい。
高速動作している状態では、どのようなOT方式にしても、相当量の惰走距離は発生
します。

◆1.6

4-15 原点復帰

4-15-1 原点復帰方式

軸設定ROMSWの「C相原点復帰」で、以下の2通りを選択できます。

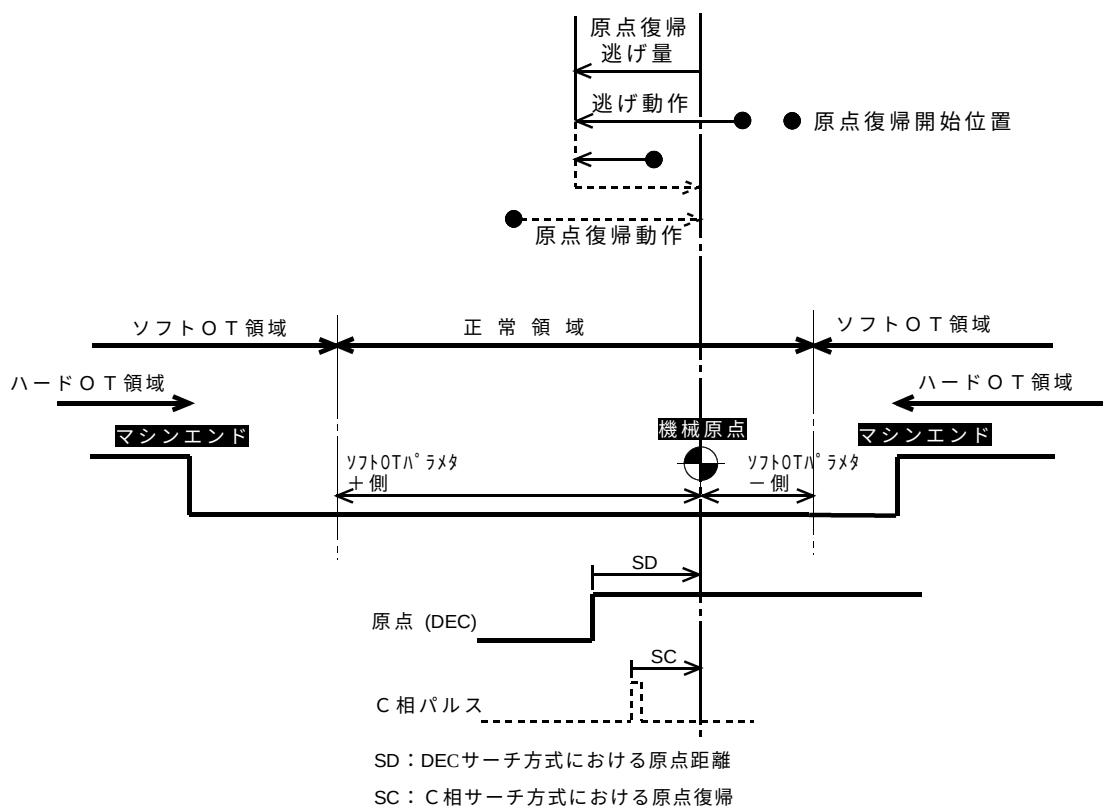
ROMSW設定

C相原点復帰	原点復帰方式	簡単な説明
有効	C相サーチ方式	エンコーダやスケールのC相（Z相）パルスを基準に原点を決めます。DECサーチより高精度です。
無効 （デフォルト）	原点サーチ方式 （DECサーチ）	原点（ORG/DEC）信号のエッジを基準に原点を決めます。センサーの経年変化でずれることもあります。

動作の詳細は、機能編8.「サーボパラメタ」の8-5「原点復帰の動作」を参照下さい。

4-15-2 軸ストロークと機械原点のようす

軸のストロークに関連して、マシンエンド、ハードOT、ソフトOT、機械原点、原点（ORG/DEC）信号、C（Z）相、逃げ動作の関係は、以下の図のようになります。



4-15-3 疑似アブソ

SLMシリーズでは、電源断の時に、アブソ位置をメモリーに記憶します。再投入されたときは、アブソ位置をその値にセットアップします。電源断中に軸が動いていなければ、疑似的にアブソエンコーダのシステムになります。ただ、電源断中に軸が動いた場合は、実際の位置と記憶の位置は異なります。

4-15-4 全軸原点復帰の逃げ動作

全軸原点復帰の時には、「疑似アブソ」の位置をチェックして、原点近くにある場合には、一度ストロークの中央の方向へ動いてから、原点側へ原点復帰動作をします。この機能により、オペレータや上位シーケンサから、機械位置にかかわらず、「全軸原点復帰」の指令がかけられます。

注意 電源断の間にメカが大きく動かされている場合には、この動作が保証できません。
オペレータが機械位置をチェックして、原点復帰して下さい。

動作の詳細は、機能編 8. 「サーボパラメタ」の 8-5 「原点復帰の動作」を参照下さい。

4-15-5 全軸原点復帰シーケンス (順位)

全軸原点復帰のときに原点復帰をおこなう順番を軸毎に設定できます。(サーボパラメタ)
普通は、上下軸が先に原点復帰し、その後水平軸が動くようなシーケンスが安全です。
それぞれの機構の条件に応じて、設定できます。

サーボパラメタの原点復帰順位

数字の小さい順 (0→1→2...) に原点復帰を行います。

例 「Zの順位=0 X, Yの順位=1」: Z軸→X軸, Y軸の順番で原点復帰

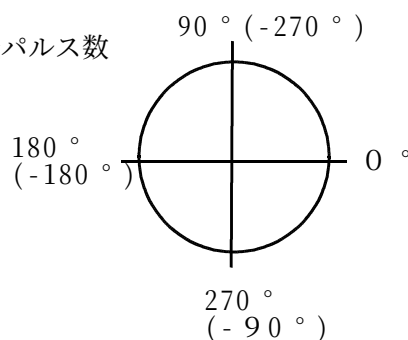
4-16 軸制御に関するその他の機能

4-16-1 周回処理(回転軸)

無限回転軸では、1周で同じ所にもどるので、座標値を1周内の値にもどす必要があります。
座標値は以下の範囲となります。

-(メカ機構一回転パルス数) < 座標値 < メカ機構一回転パルス数

ROM SWでPG分周値やギア比などを全てを考慮した上で、「メカ機構1回転あたりのパルス数」を設定します。(0以外の値を設定した場合のみ、その値で各座標値の周回処理を実行) リジットアップやねじ切りオプション機能使用時の主軸及び、実回転数ステータスが必要な主軸に対しては必ず設定します。



◆1.2

4-16-2. 回転速度指令 SPIN/G120

定速回転を指令する命令です。無限回転軸(ROM SW設定)に対して有効です。
回転数(RPM)は、機械軸での回転数です。(メカ1周パルスのパラメタ設定が必要)

命令コード「SPIN」/「G120」につづいて、回転軸を指定します。
軸指定には回転速度を0.1RPM単位で指定します。
さらに「OVR1」を付加する事により、回転動作にオーバーライドがかかります。(指定しない場合は、オーバーライドはかかりません)(G120では、OVR1は不可)

【テクノコード】

SPIN Z1000;	100rpmでの定速回転(オーバーライドは無効)
SPIN Z1000 OVR1;	100rpmでの定速回転(オーバーライドは有効)
SPIN Z0;	回転速度指令が0rpm。(停止)

【Gコード】

G120 Z1000;
Gコードでは、OVR1は対応していません。

定速回転動作の開始/停止時には、PTP時定数による直線加減速がかけられます。
移動中の軸に、回転指令をおこなうとプログラムエラーとなります。
また、回転動作中の軸に対しては、他の動作指令はプログラムエラーとなります。回転動作停止後は、再び他の移動命令(位置決め)が可能となります。
回転動作の実行中に再度回転指令を行い、回転速度を変更する事は可能です。
尚、回転動作の開始(または停止)指令時は、指令回転数への到達(または停止)を待たずに、次のステップが実行されるので、必要に応じてドウェル(タイマー)命令等を使用して下さい。

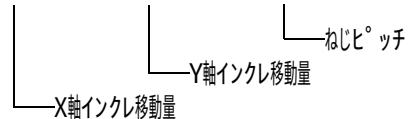
4-16-3 ねじ切り機能(同期送り)

(計画中)(専用化)

位置ループ制御型主軸(第5軸)のフィードバックデータから、X/Y軸の送り量を計算して同期指令する事により、ねじ切り動作を行います。

【Gコード】

G32 X±..... Y±..... F..... ;



- X Yどちらか一方の指令で平行ねじ、両方の指令でテーパねじとなります。
- X Y両軸動作の場合、ピッチ計算の基準軸は、移動量が多い方の軸となります。
- ねじピッチ F は、主軸 1 回転あたりの基準軸の移動量を指定します。
- G32 命令の実行時は、あらかじめ主軸の回転動作が実行されているものとします。主軸停止状態で G32 が指令された場合は、プログラムエラーとなります。
- X 軸、Y 軸の移動は、主軸のフィードバックに対する、比例分配制御によって行います。また、動作開始のタイミングを、主軸の C 相位置を基準にする事により、繰り返し時の切り込み位置を一定とします。ただし G32 命令が連続する場合、2 回目以降では主軸の C 相検出はせず前ブロックに継続して動作します。
- 同じねじ切りに対して、切り込み量を変えて繰り返しねじ切り動作を行う場合、その途中で主軸回転数 (S 値) を変更してはいけません。
- ねじ切り動作中、一時停止は無効となります。

◆1.9

4-16-4 同一指令2軸制御 <オプション>

サーボの第4軸目がX軸(第1軸)の同期軸として動作します。X軸に対する全ての動作指令(プログラム指令、動作コマンド、手動操作)に対して、第4軸がX軸と同期して動作します。通常は、平行軸などで使用します。

原点復帰完了で、お互いの2軸が正確に平行になります。その後は、手動/自動ともに常に一緒に動きます。

お互いのC相位置にずれがあっても、各軸のサーボパラメタの原点距離で調整して、正確な平行の一にします。

原点復帰の設定や注意点は、「ユーザズマニュアル」<試運転・調整編 6-11-4.同一指令2軸制御での原点復帰>(TB00-0800H)を参照して下さい。

4-16-5 PGパルス任意分周機能

ROMスイッチ設定により、各軸モーターのエンコーダフィードバックパルスが、任意の値で分周可能です。分周値は各軸毎に分子/分母の値で、分子/分母とも1～32767の値です。従来では、モーター側での分周設定が不可能であったり、変速機のギア比が割り切れないなどの理由により、1パルス＝1μmの様な切りの良い値にする事ができなかったシステムに対して、本機能によりそれが可能となります。例として、モーターのエンコーダが2000パルス/回転（固定）で、ボールネジピッチが＝5mm、（変速機は無し）の場合、分周値を5/8に設定する事で1パルス＝1μmのシステムとなります。

$$5[\text{mm/rev}] \div (2000[\text{パルス/rev}] \times 4 \times 5 / 8) = 0.001[\text{mm/パルス}]$$

詳細は、ROMSWマニュアル4-4「軸設定ROMSW」を参照下さい。

◆1.2

4-16-6 接線制御軸機能

<オプション>

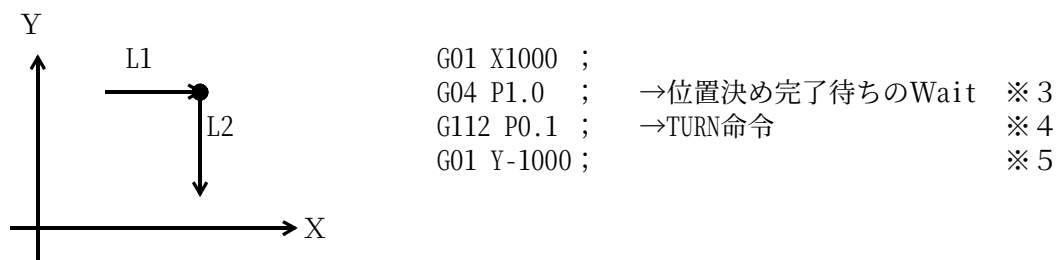
生地や板のカッティングの場合、X/Y軌跡の方向に常に歯先を向ける場合があります。本機能により無限回転軸であるZ軸(第3軸)が、XY平面上におけるX/Y軸の合成移動方向に、自動的に回転動作します。自動運転/手動運転ともに、常時この機能は有効です。運転プログラム(テクノコード/Gコード)で使う方法とPC通信LiBのコマンドで使用方法があります。

a. 接線制御用プログラムコード

G111/STND	接線制御無効 ※1
G100/STNE	接線制御有効 ※1
G112(P□□)/TURN(TIM□□)	Z軸回転命令 P□□(TIM□□)は、回転後の待ち時間(秒) 0.1～999.9秒

※1 モーダルな命令です。(システム起動時は無効です。) 有効/無効の変更はプログラム終了後も継続します。

b. 運転プログラム例



- ※3 TURN命令(G112)の前には、ドゥエル(G04)を入れることを推奨します。
- ※4 サーボ系の遅れを考慮して、Waitを入れることができます。
- ※5 TURN命令でZ軸の移動位置を算出するため、TURN命令の直後にX、Y軸の移動命令が必須となります。G112の次にX/Y関連の以下の命令がない場合は、プログラム実行エラーとなります。

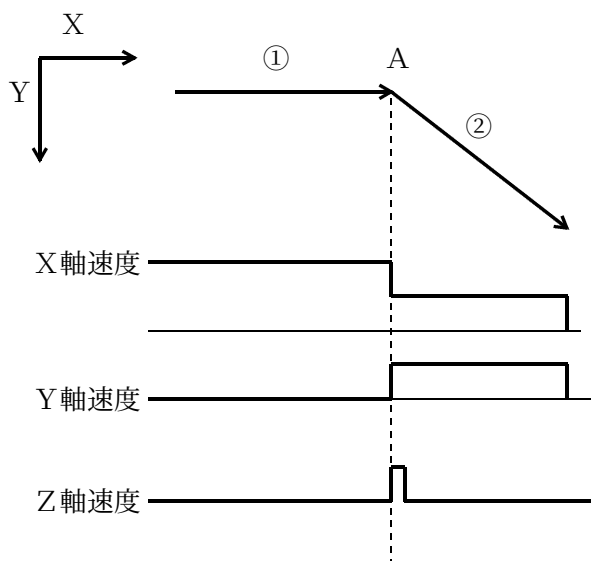
```
PTP/G91G00
LIN/G91G01
CIRR/G91G02(7°リ解円弧)
CIRL/G91G03(7°リ解円弧)
SLIN/G91G31
```

TURN命令の直後に上記命令を指定できない場合は、TURN命令にX、Yを付加することにより接線を指定することができます。この機能によりTURN命令の直後に移動命令以外の命令を指定することもできます。

※6 Z軸はなるべくサーボモータを使用下さい。
パルスモータの場合、動作がなめらかでない場合があります。

c. LIN命令(X/Y)に対するZ軸(3軸)の動作

LIN命令(G01)のX、Yのベクトル方向へZ軸(第3軸)を回転させます。



Z軸は、A点で「接線制御上限速度」(パラメタ)で回転します。

一般には位置ループの特性により速度の変化やZ軸の回転速度の変化は曲線になります。

d. ジョグ送り(手動送り)の時にも接線制御します。

例) **+X** キー
↓
Z軸 +X方向へ回転
X軸 手動送り

e. 接線制御の有効(G110/STNE)/無効(G111/STND)

G110/(STNE)の指令で接線制御有効(モジュール)となり、それ以降のX/Y移動に対してZ軸を自動追従させます。

G111/(STND)の指令で接線制御を解除します。

f. TURN命令 (G112)

直角に折れ曲がるようなカドの部分では、TURN命令を使います。



X/Y軌跡がなめらかに変化する軌跡であれば、上記機能は不要です。

g. 関連するサーボパラメタ

Z軸のサーボパラメタが特殊な意味になります。

Z軸サーボパラメタ (表示上の名称)	サーボパラメタ名称 (正式名称)	意 味
十方向ソフトリミット	接線制御上限速度	接線制御によるZ軸回転速度 α 上限値(pps)
一方向ソフトリミット	接線制御基準位置	X Y平面にて+X方向へ動く時のZ軸の アブソ位置(パルス)

◆1.5

h. 関連する ROM SW

θ 軸(第3軸)のROM SW

■メカ機構一周パルス

θ 軸が360° するパルス量

■指令極性

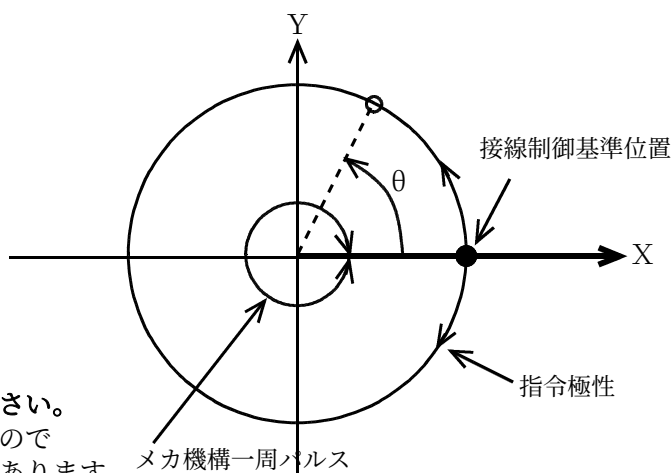
X/Yベクトルとθ 軸の+/-方向
の関係

■Z軸のOTチェック機能

必ず無効にして下さい。

※Z軸の上記のROMSWは必ず設定して下さい。

※θ 軸はX/Yベクトルに近回りで動作するので
常に一定方向に回転する動きになることがあります。



i. PC通信ライブラリーにおける使用方法

<PC通信ライブラリー>

コマンド送信関数に、Z軸接線制御ON/OFFコマンド **REQ_TLINE**が追加となります。

本コマンドにより、接線制御のON/OFFが切り換えられます。

尚デフォルトは接線制御OFFとなっています。

<付加パラメータ構造体>

```
typedef struct {
    unsigned short tlinesw;    Z軸接線制御機能スイッチ ..... 0=OFF, 1=ON
} TLINE_SW;
```

通信ライブラリーデータ受信関数のステータス要求DAT_STATUSで得られる、ステータスデータ構造体のメンバー「ステータス情報」(.Status)の **S_TANG**ビットがONの時、接線制御がON状態である事を示します。

4-16-7 センサーラッチ機能（スキップ機能） <オプション>

本機能は特殊な直線補間を実行します。センサーラッチ入力（CN2 80番）がONすると、その瞬間の機械位置（論理座標系及びアブソ座標系）を制御軸数分メモリーにセットします。マクロ変数#500□、#510□です。詳細は、機能編6-4-11「センサーラッチポジションデータ」を参照下さい。

また動作としてはプログラム運転の場合、同信号入力の論理ONにより移動指令の残りを中断して次のステップに進みます。（信号の論理はROMSW「軸設定」のHIOのL/Hで選択）
（スキップ機能） ◆2.1

通信コマンドによる動作の場合は、プログラム運転と同様にスキップするか、最後まで移動するか（スキップキャンセル）を指定できます。

センサーラッチ位置情報は、マクロ変数や通信コマンドによって読み出す事ができます。

<運転プログラムコード>

テクノコードでは“SLIN”、Gコードでは“G31”の指定により、本機能を実行します。基本的な動作方式及び制約事項は、テクノコードの“SLIN”は従来の“LIN”に、Gコードの“G31”は従来の“G01”に等価です。（ただしG31はノンモーダル指令となります。）

4-16-8 コンベア追従送り 同調/同期追従送り <オプション> (計画中)

ベルトコンベア上などを移動するワークに対して、同調／同期追従制御が有効です。

同調送り (TUN)	ベルトコンベア上のワークに対して、位置と速度を合わせます。
同期追従送り (DLIN)	ベルトコンベア上のワークに対して、同じ速度で動きます。

TUN X(1 m) Y(LM) ;

無限回転軸であるY軸を、あらかじめSPIN（またはSPIN2）命令により、回転動作させておきます。TUN命令により、両軸の論理座標系機械位置が指定した（1m, LM）に到達する様に、Y軸のフィードバックに同調してX軸を動作させます。

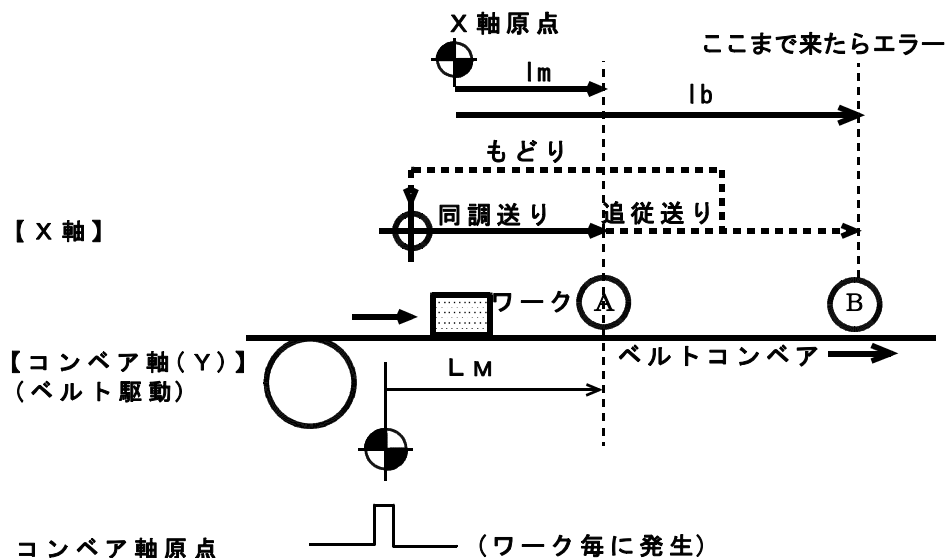
指定点を通過した時点で分配完了となり、次のステップへ進みます。

尚、TUN命令を開始する時点で、既にX軸の座標が指定位置を越えていた場合、プログラムエラーとなります。また、Y軸の座標が指定位置を越えていた場合は、Y軸が周回処理によって1度ラウンドアップした後、再度到達する指定位置をYの目標点として動作します。

```
DLIN  X(1b)  TIM□□□  DO*****;
```

無限回転軸であるY軸を、あらかじめSPIN（またはSPIN2）命令により、回転動作させておきます。DLIN命令によりTIMで指定された時間だけ、Y軸のフィードバックに同期してX軸を移動させます。この間に論理座標系のX軸機械位置が、指定した座標値(LB)に到達してしまった場合は、プログラムエラーとなります。また本命令では同時に、標準のDO命令による汎用出力信号制御を実行できます。

使用例



コンベア軸は、ワーク毎に原点パルスを出して下さい。
 一般には、回転軸座標として、コンベア軸の座標をワーク毎にセットアップします。

A点 同調を完了させるべき目標位置。(L_M)
 ここまでにX軸は加速して位置・速度を合わせます。

<プログラム例>

X軸のもどり位置でシリンダを上／下。(R00出力:シリンダ ON/OFF)
追従区間でシリンダを上／下させます。
P01入力ONで終了。

```
PTPA    X    0                ; X軸 ホーム位置へ
SPIN2   Y    600              ; ベルトコンベア起動(直線型加減速)
      }
ONR00   TIM 0.1                ; シリンダ下降(0.1秒)
      TIM 0.1                  ; 0.1秒
OFR00   TIM 0.1                ; シリンダ上昇(0.1秒)
TUN     X(Lm) Y(LM)            ; 同調

DLIN    X(POSB) ONR00 TIM0.1   ; 同期追従 シリンダ下降
DLIN    X(POSB) TIM0.1         ; 同期追従 0.1秒
DLIN    X(POSB) OFR00 TIM0.1   ; 同期追従 シリンダ上昇

PTPA    X    0                ; X軸もどり
SPIN    Y    0    JNR01        ; R01入力 ONの時コンベア停止
      ER01                    ; R01入力 ONの時強制終了
END      (サイクル運転)        ; サイクル運転(エンドレス)
```

◆1.2

4-16-9. 直径指令 <オプション>

旋盤等の機械で、X軸(切り込み軸)の指令値をワークの直径値で指令します。
運転プログラム内の数値だけが直径指令として扱われます。この場合、実際の移動は、指令値の半分です。
ワンショットPTPやサーボパラメータは半径値として扱われます。(実移動量)
また、現在位置表示画面の指令位置は直径指令値として表示され。
アブソ位置、機械位置、偏差量は半径指令値として表示されます。

例: X軸を直径指令とし、運転プログラム中で以下のような記述をした場合

```
PTP X100;
指令位置には100と表示され、アブソ位置、機械位置、(偏差量)には50と表示されます。
```

ワンショットPTPでX軸に100と命令したときは、指令位置には200と表示され、
アブソ位置、機械位置、(偏差量)には100と表示されます。

◆2.1

4-16-10. 巻線指令 <オプション>

ボビンに巻線を行う専用の命令です。(詳細はTB00-0934を参照してください)
この命令に対応するGコードは有りません。(テクノコードのみです)

巻線指令

```
REEL { トラバース軸±□ } P±□ RN±□ S±□ [ RE±□ ] ;
```

{トラバース軸±□} は次の軸の中の何れかの1軸です X、Y、Z

※ A軸は主軸固定として使用するため、トラバース軸としては指定できません。

[] ... 囲まれた項目が省略できることを示します。

・トラバース移動幅

トラバース幅が+の場合、最初に+方向に移動して、往復運動を行います。

-の場合、最初に-方向に移動して、往復運動を行います。

・P:ピッチ幅

主軸一回転当たりのトラバース軸の移動量を指定します。(パルス/REV)

・RN:総巻数

総巻数(主軸回転数)を指定します。

巻き数が+の場合、主軸は+方向に回転します。

-の場合、主軸は-方向に回転します。

- ・ S：主軸速度
主軸の回転速度を指定します。
- ・ R E：巻き終わり処理選択
巻き終わりの処理の方法を指定します。
設定内容は以下の通りです。
0：自然終了（デフォルト）
1：トラバース始端で終了
2：トラバース終端で終了

【動作】

主軸が指定された速度で回転し、トラバース軸は主軸**指令**に同期して往復動作します。
主軸速度には、サーボパラメータの「PTP時定数」による加減速がかかります。
トラバース移動は主軸指令に同期しますので主軸と同様の加減速がかかります。

◆1.1,1.2,1.6,1.7

4-17 主軸機能 <オプション>

主軸機能は、指令された一定速度でサーボモータやインダクションモータを回転してワーク軸やツール軸として使用するものです。

方 式	内容・特徴	ハード	指 令 方 式	
			Gコード	Tコード
オープンループ方式 (アナログ速度指令) 主軸回転指令	PG FB 不要。 一般の主軸	第5軸	S5 々 (RPM) M03 (正転) M04 (逆転) M05 (停止) ----- G120	SPIN命令
2ビット出力信号 制御方式 (主軸 ON/OFF)	I/O 信号のみ アナログI/F不要。	一般出力 SOUT0 SOUT1	S0 停止 S1 速度 1 S2 速度 2 S3 速度 3 +M03/M04/M05 ----- G120	

※ 旋盤の主軸は、一般にはオープンループ方式です。ネジ切りのためには、主軸 PG FB が必要となり、専用化対応となります。

4-17-1. オープンループ方式

主軸操作は、“コマンド”、“S指令”、“テクノコード”、“Gコード”から行い、最後に指令した命令が有効となります。

<G言語運転プログラムによる指令>

Gコードプログラムの主軸回転速度指令として、Sコードが有効となります。

書式は、Sに続けて回転速度[rpm]を指定します。【例】S300；

実際の主軸動作は、M03(正転)／M04(逆転)／M05(停止)の各命令により実行されます。尚、主軸回転の開始（または停止）指令時は、指令回転数への到達（または停止）を待たずに、次のステップが実行されるので、必要に応じてドウェル命令等を使用して下さい。

Mコード出力あり（基本ROMSW：主軸指令Mコード出力をチェック）のときには、M0□実行時にそのMコード出力(8bit)も動作します。

Mコード出力時はFIN待ちを行います。FINが返るまで、新たなMコード発行や移動コマンド等の発行は出来ません。

「アナログ指令」オプションが有効の時、第5軸に対して“G120”命令を指令すると主軸動作を行います。指定した主軸速度が主軸最高回転数（サーボパラメータのPTP速度）を越えた場合、「プログラム実行エラー」になります。

※ “G120”命令については、「本マニュアル(TB00-0800E) 4-16-2 回転速度指令 SPIN／G120」を参照下さい。

<サーボパラメータ>

第5軸のサーボパラメータに以下の3項目を設定します。
尚、これら3項目以外の第5軸用パラメータについては、全て不要
(いくつに設定してあっても無関係)となります。

①主軸最高回転数 (PTP速度.PtpFeedに設定)

主軸の最高回転数を指定します。

設定範囲は、10～100000 [rpm] です。

◆2.3

本設定値を越える値が指令された場合、主軸回転数は本設定値でクランプします。

②主軸指令電圧 (D/A出力ゲイン.Gainに設定)

主軸最高回転数で主軸を回転させる時の、アナログ指令電圧の値を設定します。

主軸回転速度にマイナス値が指定された場合は、対応するアナログ指令電圧もマイナス値になります。設定範囲は、1～10 [V] です。

③主軸時定数 (PTP時定数.Dxに設定)

主軸の停止指令 ←→ 最高回転指令の加減速時間を指定します。

設定範囲は、「ROMSWの制御周期の値」～「6400」 [msec] です。

※パラメータと出力電圧の関係

・Sで指定した回転速度と出力電圧、各パラメータは以下の関係にあります。

$$\text{出力電圧[V]} = \frac{\text{サーボパラメータのD/Aゲイン} \times \text{Sで指定した値}}{\text{サーボパラメータのPTP速度 (主軸の最高速度)}}$$

<PC通信ライブラリによるアナログ指令出力>

PC通信ライブラリ動作コマンドREQ_SPREVSETとREQ_SPCMNDの発行で、アナログ指令制御が可能です。REQ_SPREVSETで、主軸速度を設定します。

動作コマンドREQ_SPCMND(パラメータ=0)の発行では、常にS0相当の出力となります。

ただし、主軸速度はモータ情報として記憶されていますので、再度REQ_SPCMND(パラメータ=1または2)を指令すると記憶していた主軸速度に対応する出力を行います。

逆に、あらかじめREQ_SPCMND(パラメータ=1または2)を発行しておき、その後REQ_SPREVSETで、主軸速度を変更する事もできます。

<テクノ言語では>

「アナログ指令」オプションが有効の時、第5軸に対して”SPIN”命令を指令すると主軸動作を行います。指定した主軸速度が主軸最高回転数(サーボパラメータのPTP速度)を越えた場合、「プログラム実行エラー」になります。

※ ”SPIN”命令については、

「本マニュアル(TB00-0800E) 4-16-2 回転速度指令 SPIN/G120」を参照下さい。

4-17-2. 2ビット出力信号方式

<G言語運転プログラムによる指令>

Gコードプログラムの主軸回転速度指令として、Sコードが使えます。

書式は、Sに続けて0～3のコードを指定します。 【例】S1;

S命令は、モータ情報です。(一端指令すると記憶しています)

実際の出力は、M03(正転)/M04(逆転)/M05(停止)の各命令により実行されます。また、M03(04)出力後に、S1→S2→S3などと指令することで主軸の速度を変えられます。

2bit出力に関しては、M03とM04では差がありません。

尚、主軸回転の開始(または停止)指令時は、指令回転数への到達(または停止)を待たずに、次のステップが実行されるので、必要に応じてドウェル命令等を使用して下さい。

主軸速度指令 S命令(モータル)	M03/04実行時		M05実行時	
	SOUT1出力	SOUT0出力	SOUT1出力	SOUT0出力
0:S0	OFF	OFF	OFF	OFF
1:S1	OFF	ON	OFF	OFF
2:S2	ON	OFF	OFF	OFF
3:S3	ON	ON	OFF	OFF

Mコード出力あり(基本ROMSW:主軸指令Mコード出力をチェック)のときには、M0□
実行時にそのMコード出力(8bit)も動作します。

Mコード出力時はF I N待ちを行います。F I Nが返るまで、新たなMコード発行や移動コマンド等の発行は出来ません。

<PC通信ライブラリによる2ビット出力>

PC通信ライブラリ動作コマンドREQ_SPREVSETとREQ_SPCMNDの発行で、SOUT□出力の制御が可能です。

REQ_SPREVSETで、S0～3のモーダル選択をします。

動作コマンドREQ_SPCMND(パラメータ=0)の発行では、常にS0相当の出力となります。

ただし、S1～S3の選択はモーダル情報として記憶されていますので、再度REQ_SPCMND(パラメータ=1または2)を指令すると記憶していたS□に対応する出力を行います。

逆に、あらかじめREQ_SPCMND(パラメータ=1または2)を発行しておき、その後REQ_SPREVSETで、S0～3の出力を変更する事もできます。インバータの、低速/中速/高速などの選択に応用できます。

Gコードプログラム		PC通信ライブラリ		SOUT1	SOUT0	主軸動作指定
Mコード	Sコード	REQ_SPCMND	REQ_SPREVSET			
M05	S0	パラメータ=0	パラメータ=0	OFF	OFF	停止
M03/M04	S1	パラメータ= 1：正転 2：逆転	パラメータ=1	OFF	ON	速度1
M03:正転	S2		パラメータ=2	ON	OFF	速度2
M04:逆転	S3		パラメータ=3	ON	ON	速度3

<テクノ言語では>

テクノ言語からは2ビット出力は指令不可です。

4-17-3. 主軸制御と無限回転軸(一般の軸)に対するSPINの違い

		主軸制御		無限回転軸に対する
		アナログ指令	2ビット出力	SPIN
制御方式		DA指令(FBなし)	インバータ速度 選択 S0～S3	パルス指令の位置ループ
対応軸 (2ビット出力ではI/O接続)		第5軸(オプション)	一般出力SOUT0/ SOUT1でI/O出力	全ての軸
オプション選択		主軸アナログ指令	主軸ON/OFF	(標準) ※1
プログラム 指令	Gコード	S指令+ M03～M05 (主軸オーバーライド) ※3	S0～S3+ M03～M05	G120 (オーバーライドは無し)
		G120※2 (主軸オーバーライド) ※3	G120	
	テクノコード	SPIN(アナログ)※2 (主軸オーバーライド) ※3 (OVR0/1は無視)		SPIN OVR1でオーバーライド
セッティング PC表示	指令位置	主軸指令(RPM)		指令位置(パルス)
	機械位置	0		FB有効 : 機械位置(パルス)
				FB無効 : 指令位置(パルス)

※1 無限回転軸の場合は、ROM SWで「1回転パルス数」の設定が必要です。

※2 主軸オプション有で第5軸に対してSPINやG120指令をした場合は、内部的には主軸指令として動作します。同時にM3/M4/M5のMコードが動作します。 ◆2.3

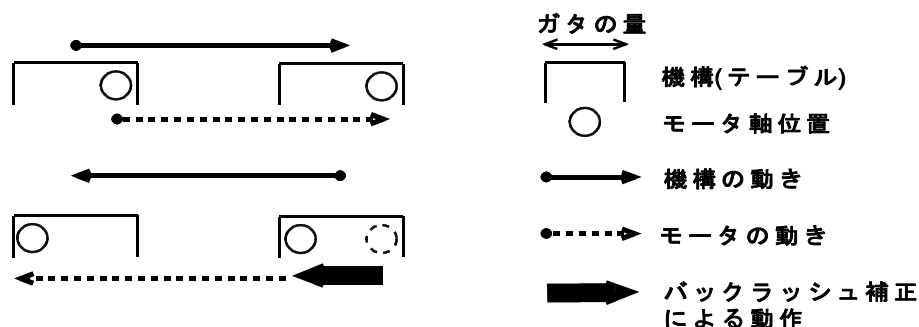
※3 ROM SWの基本パラメータ「主軸オーバーライド」が有効設定の時には主軸指令にオーバーライドがかかります。

4-18 補正機能

4-18-1 バックラッシュ補正機能

機械系に、いわゆる「ガタ」(不感帯)がある場合には、「バックラッシュ補正」のパラメタに「ガタ」に相等するパルス量を設定して下さい。

【バックラッシュ補正の概念図】



バックラッシュ補正は、軸の移動方向が変わるたびに、その方向へ加わります。

〈最初のバックラッシュ補正〉

◆2.1

原点復帰完了後に、「原点復帰中の最後に原点に向かって動いた方向と逆の方向」に動いたときに、バックラッシュ補正がかかります。

原点復帰未完では、無効です。

アブソリュートエンコーダ付きの軸にはバックラッシュ補正は使えません。

4-18-2 工具長補正機能及び摩耗補正機能 <オプション>

プログラムコード

G43 H□□; / THSET□□;

補正テーブル番号

により、工具長補正及び摩耗補正を実行します。

すでに補正がかかった状態で、新たな補正番号が指令された場合、現在の補正量と新たな補正量の差分で補正がかかり、常に最後に指令された補正番号の補正がかかった状態になります。

プログラムコード

G49; / THOFF;

により、工具長補正及び摩耗補正をキャンセルします。

補正テーブル

T No.	工具長補正量	摩耗補正量
0	-1000000000 ~ 1000000000	-1000000000 ~ 1000000000
.	.	.
.	.	.
.	.	.
1 9	-1000000000 ~ 1000000000	-1000000000 ~ 1000000000

指定する補正テーブル番号は工具長補正と摩耗補正で共通となりますので、ご注意下さい。

○工具長補正

指定された補正テーブル番号の工具長補正量を現在の論理座標値に加算します。

軸の移動は行いません。

工具補正が指定された状態で、原点復帰動作を行うと、動作完了時に論理座標値が再セットアップされます。

本補正を実行した後の最初の移動命令は必ずアブソ指令にして下さい。

○摩耗補正（現在未完成で、対応していません）

指定された補正テーブル番号の摩耗補正量の分だけ、軸の移動を行います。

ただし、論理座標系をシフトする事により移動前と移動後で論理座標値は変化しません。

原点復帰動作を実行した場合、摩耗補正は自動的にキャンセルされます。

※ 摩耗補正機能は、現状では対応していません。

工具長補正/摩耗補正のプログラム例

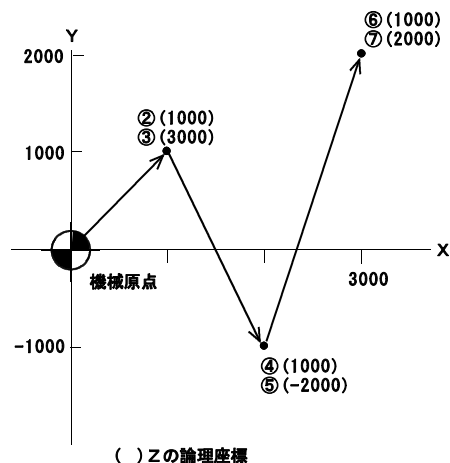
```

① G 90 G00 X0 Z0 ;
② G 91 G01 X1000 Z1000 F1000 ;
③ G 43 H0 ;
④ G 90 G00 X2000 Z1000 ;
⑤ G 43 H1 ;
⑥ G 90 G00 X3000 Z1000 ;
⑦ G 49 ;
    
```

例 1

HN0.	工具長補正	摩耗補正
0	2 0 0 0	0
1	- 1 0 0 0	0

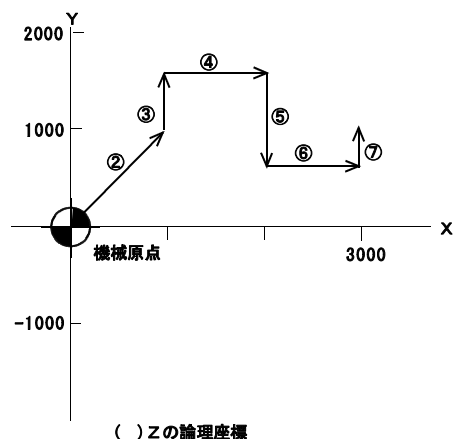
動作 ステップ°	論理座標		アブソ座標(メカ)	
	X	Z	X	Z
①	0	0	0	0
②	1000	1000	1000	1000
③	1000	3000	1000	1000
④	2000	1000	2000	-1000
⑤	2000	-2000	2000	-1000
⑥	3000	1000	3000	2000
⑦	3000	2000	3000	2000



例 2

HN0.	工具長補正	摩耗補正
0	0	5 0 0
1	0	- 5 0 0

	論理座標		アブソ座標(メカ)	
	X	Z	X	Z
①	0	0	0	0
②	1000	1000	1000	1000
③	1000	1000	1000	1500
④	2000	1000	2000	1500
⑤	2000	1000	2000	500
⑥	3000	1000	3000	500
⑦	3000	1000	3000	1000



◆1.5,1.6

【注意】

同一行で工具長補正/摩耗補正命令と汎用入力制御（JR命令など）を設定した場合でも、論理座標のセットアップは必ず実行されます。入力制御によるスキップ（JUMP）は、ききません。座標のセットアップを行った後に汎用入力制御(Exit、スキップ、Wait、Stop)による処理を行います。

例 THSET H1 JR00 ;

R 0 0 入力にかかわらず、THSETは内部処理されます。

※ 工具長補正では、軸移動が発生しないので、スキップ(J R)指定は実質的に無効です。

4-18-3. 工具径補正機能 <オプション>

詳細については、「SPX径補正機能説明書」TB04-1632を参照ください。
使用上の注意点などがあります。

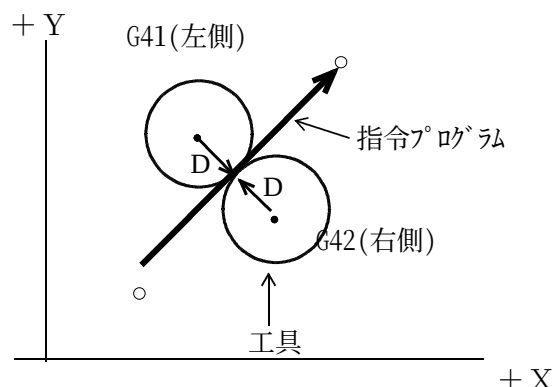
(1) 径補正指令コード

■ Gコード (G40、G41、G42)

【Gコード例】

```
G41 (G01) D__ X__ Y__      ;左側補正
{
G42 D__ X__ Y__            ;右側補正
}
G40                          ;径補正キャンセル
```

G40	径補正キャンセル
G41	径補正 左側
G42	径補正 右側
D	工具径補正テーブルNO. (2けた)



- a) Dの値はモーダルです。
(省略可、プログラムモーダル※)
- b) Dの指定範囲は0～19です。
- c) 工具径補正テーブルの補正量(+値)は、あらかじめセットしておきます。
- d) 径補正は、X/Y平面にておこないます。(2次元)
- e) G40、G41、G42は、単独でも、またG00、G01と組み合わせても使用できます。
- f) 円弧指令ではアドレスDの値は第4軸中心指定です。径補正値の設定はできません。
モーダルでの設定となります。
- ※ 一回のプログラム実行の中でモーダルです。
前回のプログラム実行で指令したDの値は引き継ぎません。
プログラム起動時は、デフォルト値(D00)になります。

■ テクノコード (DC、DL、DR)

【テクノコード例】

```
LIN X__ Y__ F__ DL__      ;左側補正
{
LIN X__ Y__ F__ DR__      ;右側補正
}
DC                          ;径補正キャンセル
```

DC	径補正キャンセル
DL	径補正左側+工具径補正テーブルNO. (2けた)
DR	径補正右側+工具径補正テーブルNO. (2けた)

- a) DL/DRで工具径補正テーブルNO.を指定します。
(DL/DR指令時は必ず工具番号を指定してください。)
- b) DL/DRの指定範囲は0～19です。
- c) 工具径補正テーブルの補正量(+値)は、あらかじめセットしておきます。
- d) 径補正は、X/Y平面にておこないます。(2次元)
- e) DC、DL、DRは、単独でも、またPTP(A)、LIN(A)と組み合わせても使用できます。
円弧指令との同時使用はできません。

4-18-4 ピッチエラー補正機能 <オプション>

各軸毎に機械原点を基点とした、一定間隔の区間毎の補正データを、パラメータに記憶させ、そのデータに従って、ボールネジの誤差を自動的に補正する機能です。
この機能は原点復帰完了後、有効になります。

【基本仕様】

補正軸	全制御軸中の任意の指定軸
補正点の数	全軸で1000点Max.
補正の基点	機械原点
補正の間隔	1000～1000000パルス
データ設定方式	アブソリュート（基点からの補正量）
補正量	0～±127パルス
補正倍率	×10Max.

【注意事項】

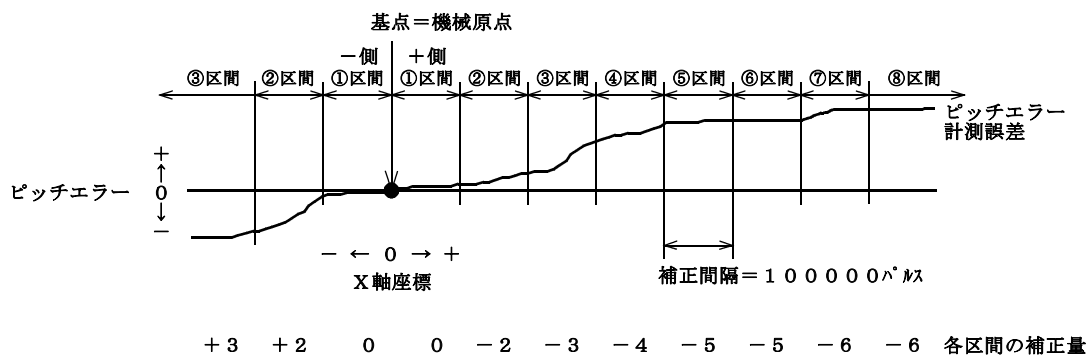
- (1) 最大補正量は、基点に対して±1270パルスとなります。
- (2) 使用するサーボの応答性やサーボ調整により異なりますが、隣合う補正量の差が大きすぎると、急激な動作となる場合があります。
- (3) 総補正点数が1000点以内であれば、軸毎の点数配分は任意です。
- (4) ピッチエラー補正を無効とする場合は、パラメータの補正倍率に、0を設定して下さい。

【パラメータ】

	項 目	設 定 値
各 軸	補正倍率	0～10
	補正間隔	1000～1000000パルス
	補正データ先頭番号	0～999
	－側補正区間数	0～1000
	＋側補正区間数	0～1000
共 通	補正データ	0～±127パルス
	No.0～No.999	

【設定例】

下図の様なX軸の補正を設定する例を示します。



X軸		補正データ		X軸で使用
		No.	補正值	
補正倍率	: 1	000	3	X軸で使用
補正間隔	: 100000	001	2	
補正データ先頭番号: 0		002	0	
一側補正区間数: 3		003	0	
+		004	-2	
		005	-3	
		006	-4	
		007	-5	
		008	-5	
		009	-6	
		010	-6	他の軸で使用
		011		
		012		
		999		

◆1.2

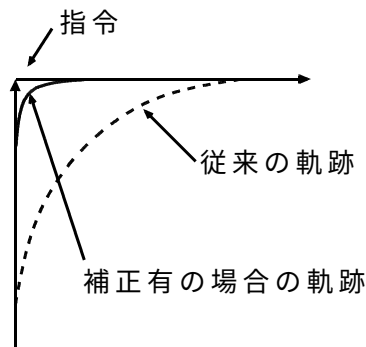
4-18-5. 形状補正(高精度輪郭制御)

<オプション>

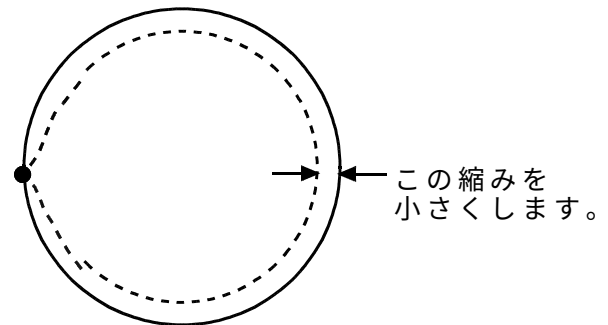
高速・高精度な輪郭制御を要求するマシンでは、形状補正による軌跡精度の向上が有効です。
これにより、サーボ系の遅れによる軌跡誤差を最小限に抑えることが可能となります。

(1) 形状補正機能の概要

a. コーナ部の内回りを少なくする



b. 円弧の縮みをなくす



(2) 形状補正の効果

サーボ系を1次遅れ系とみなして、その遅れ分を補償します。その結果、指令どおりの実軌跡となります。

速度ベクトルの変化がなめらかな場合には、最大限の効果が出ます。

従って、速度変化が大きい場合は、「プリ解加減速」を利用するか「運転プログラムの各ステップ(ブロック)を細分化して、速度Fがゆるやかに変化する」ようにして下さい。

(3) 形状補正機能の制限事項

速度が急に変わる動きでは、十分な効果が上がらない場合もあります。

また、理想的なサーボ系(1次系)と現実のサーボ系の差(機械系、速度ループ特性)から、効果の程度が変わることがあります。

(4) 使用方法 <オプション>

サーボパラメタの「形状補正係数」を設定して下さい。

形状補正係数：位置ループゲイン(K_p)の1.5倍～5倍

例	K_p	形状補正係数
	40	60～200

0の時は制御なし。

初め5倍の値を設定して、すこしずつ小さくして下さい。(軌跡を確認しながら)
1倍以下は、使用不可。

(5) 任意曲線における高精度輪郭制御の実例

【運転プログラム】

微小直線補間連続ブロック
(全432ブロック)

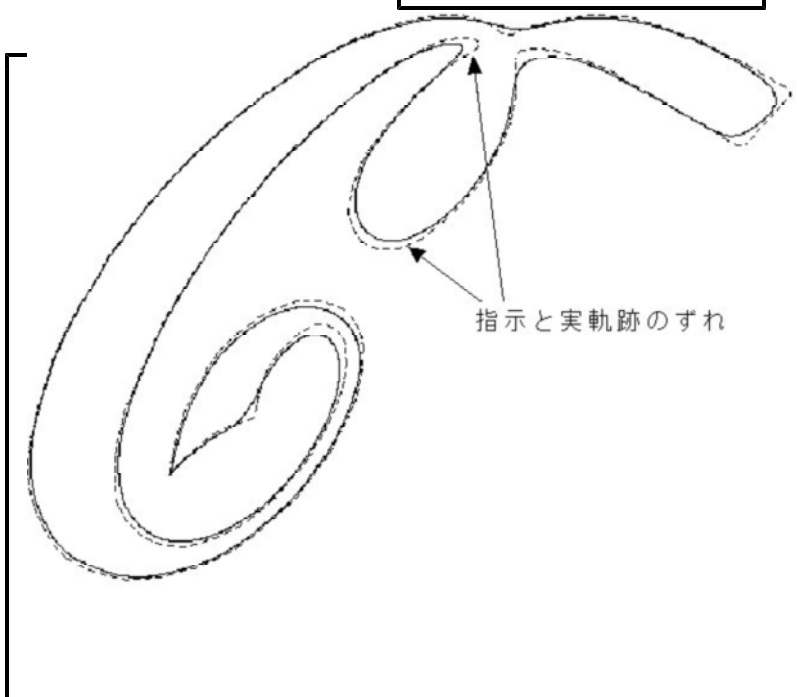
【共通動作条件】

位置ループゲイン $K_p = 30$
(XY軸共通)

```
LIN X339 Y367 F50000;  
LIN X339 Y367 F50000;  
LIN X292 Y316 F50000;  
LIN X359 Y347 F50000;  
LIN X386 Y318 F50000;  
LIN X410 Y286 F50000;  
LIN X432 Y252 F50000;  
LIN X451 Y216 F50000;  
LIN X467 Y179 F50000;  
LIN X480 Y141 F50000;  
LIN X85 Y21 F50000;  
LIN X240 Y220 F50000;  
LIN X-250 Y-433 F50000;  
LIN X-232 Y-443 F50000;  
LIN X-213 Y-452 F50000;  
LIN X-195 Y-461 F50000;  
LIN X-175 Y-468 F50000;  
LIN X-155 Y-475 F50000;  
LIN X-135 Y-482 F50000;  
LIN X-116 Y-486 F50000;  
LIN X-95 Y-491 F50000;  
LIN X-74 Y-494 F50000;  
LIN X-42 Y-372 F50000;  
END;
```

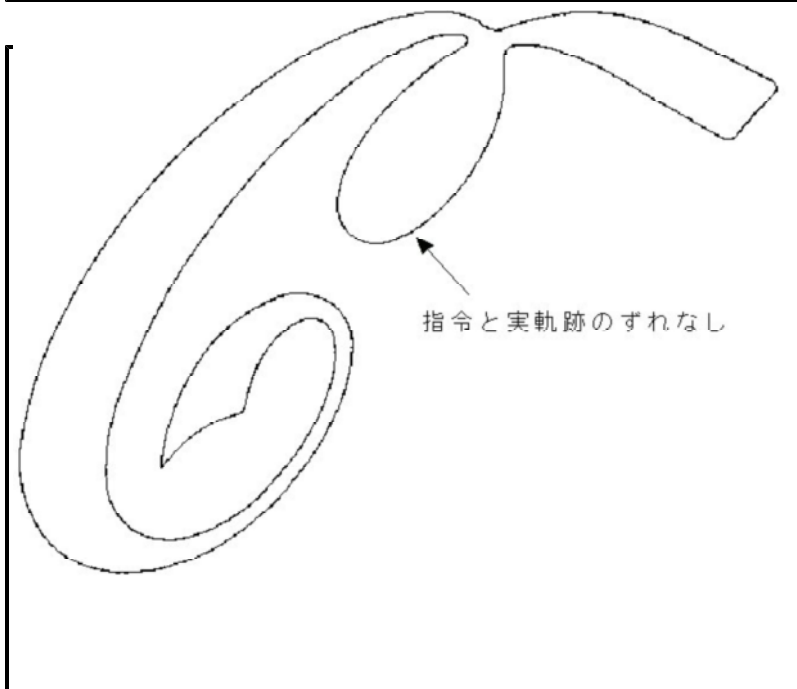
形状補正なし

- ・指数型加減速処理：有効
(時定数 = 30 msec)
- ・プリ解加減速機能：未使用
- ・形状補正制御：無効



形状補正あり

- ・指数型加減速処理：無効
- ・プリ解加減速機能：使用
(加減速時間 = 100 msec)
- ・形状補正機能：有効
(形状補正係数 = 60)



実

(6) 円弧における高精度輪郭制御の実例

プリ解加減速と形状補正を利用して、高精度輪郭制御を実現しました。

① プリ解析による加減速発生

通常、NCの内部処理では軌跡発生処理のあと、速度変化や機械のショックをやわらげるために各軸の指令に対して加減速をかけます。この加減速による遅れ要素と位置ループ系による偏差量が加わって、実際の軌跡が、コーナーの内側を回るなどの、大きな形状誤差が発生します。ただし、加減速処理を行わないと、指令の速度変化にサーボ系が追従できないという別の問題が生じます。そこで補間後の加減速処理を使用しなくても十分に追従できるようにプリ解析処理において加減速区間微小ブロックを発生させて、補間前加減速をおこないます。

【円弧動作での各軸の速度・通常】

プリ解加減速無し

補間時定数 = 30 ms

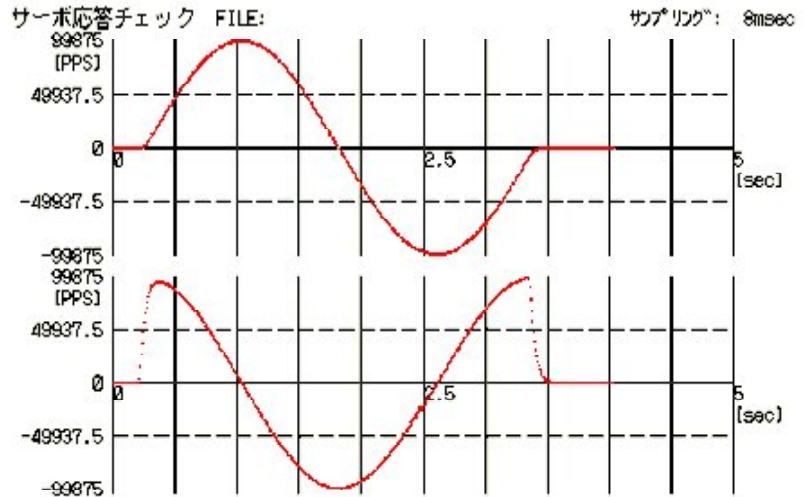
$K_p = 30$

$R = 50000 p$

$F = 100 Kpps$

でのサーボ応答

(速度指令)



【動作での各軸の速度・プリ解加減速】

プリ解加減速有り

(補間前加減速)

補間時定数 = 0

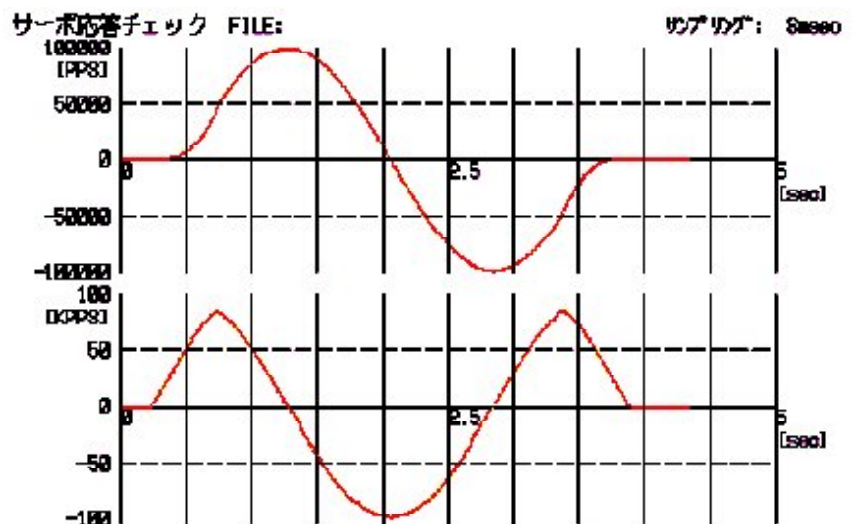
$K_p = 30$

$R = 50000 p$

$F = 100 Kpps$

でのサーボ応答

(速度指令)



② 形状補正の効果

プリ解加減速をおこなっても、位置ループ系による偏差量分の遅れのため、高速送り(位置偏差大)では軌跡誤差が顕著となります。これを0に近づける為に、コントローラ内でフィードフォワード的な補償制御をおこなうことで、位置偏差を小さくします。

当社モーションアナライザでの真円精度解析の例を示します。プリ解加減速と形状補正によって、円弧の縮小はゼロになっています

プリ解加減速無し 形状補正無し

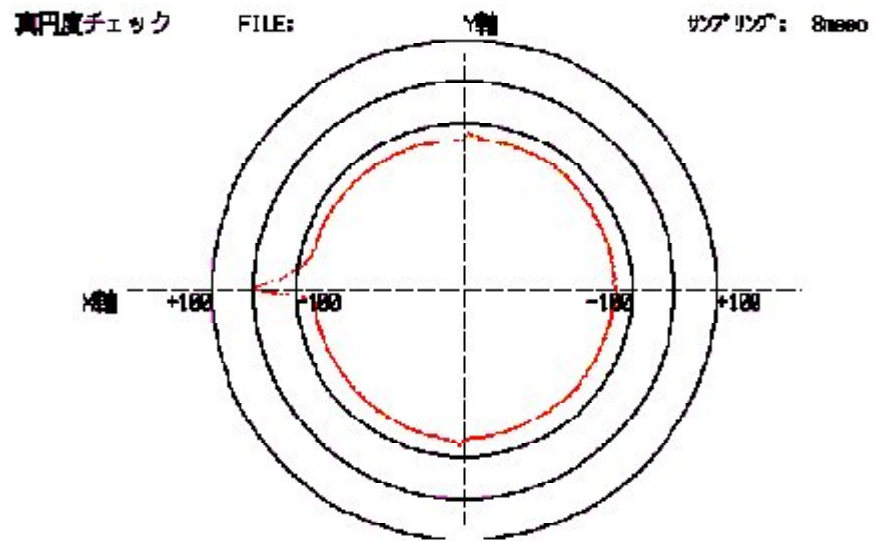
補間時定数 = 30 ms

$K_p = 30$

$R = 50000 p$

$F = 100 K_p p s$

半径縮小 = 130 p
(130 μm)



プリ解加減速有り 形状補正無し

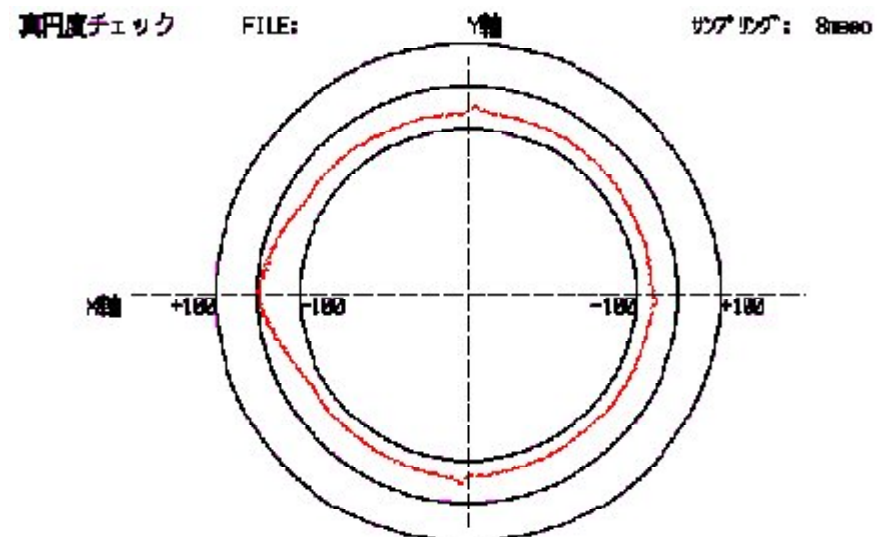
補間時定数 = 0

$K_p = 30$

$R = 50000 p$

$F = 100 K_p p s$

半径縮小 = 70 p
(70 μm)



プリ解加減速有り 形状補正有り

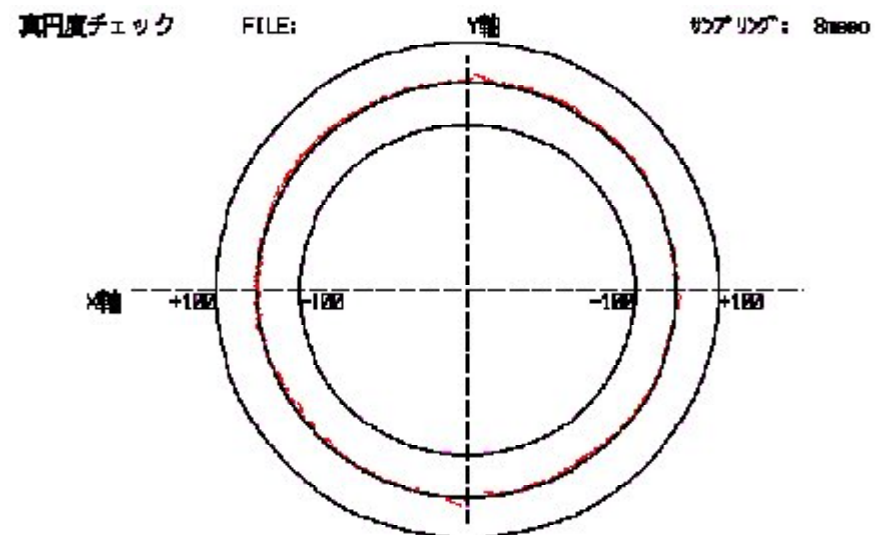
補間時定数 = 0

$K_p = 30$

$R = 50000 p$

$F = 100 K_p p s$

半径縮小 = 0



(7) プリ解加減速(プリ解析における加減速ステップの発生) <オプション>

① プリ解加減速の意味

プリ解加減速は、直線補間や円弧補間に対してスタート区間(加速区間)と停止区間(減速区間)のステップを発生させるものです。

- a. 各軸の指令速度の変化を少なくして、形状補正の効果を最大限にします。
- b. 補間後の各軸の加減速をなくしても、オーバーシュートや振動がおきないようにします。

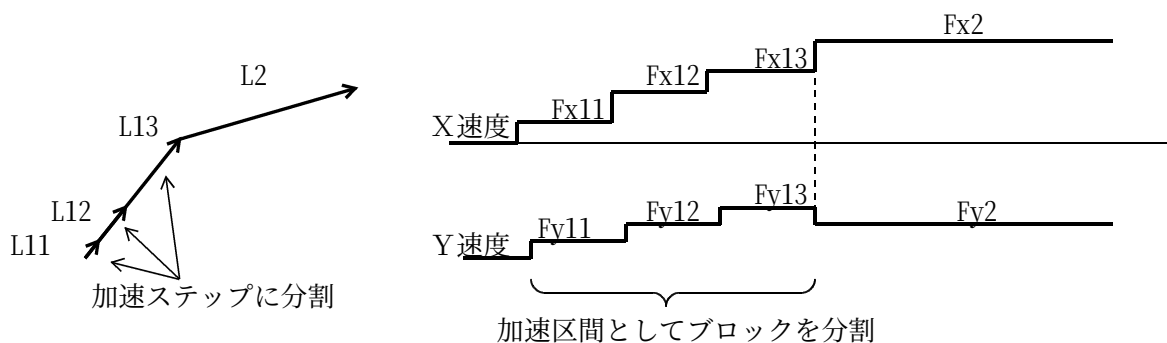
② プリ解加減速の使用上の注意

加速／減速を設定するブロック(ステップ)は必ず「プリ解加減速区間(時間)」よりも時間的に長いことが必要です。

【元の指令】



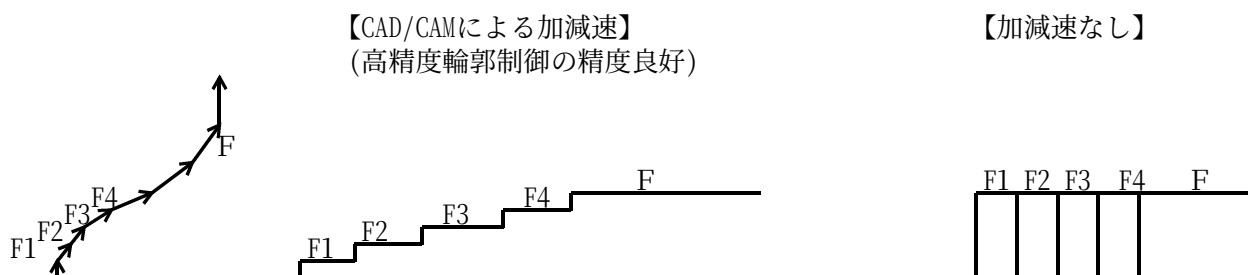
【L1を加速した例】



従って、加速／減速指令をするブロック(ステップ)では、以下を確認して下さい。
 $\text{ブロック処理時間}(L/F) \geq \text{プリ解加減速区間(時間)}$

③ 微小ブロックの処理

2項の条件式を満たさない微小な(短い)ブロックの場合は、CAD/CAMによるブロックの発生において、あらかじめ接線速度を加速／減速させることを推奨します。



この処理については、必然ではありませんが、高精度輪郭制御を効果的にするためには、必要です。

4-19 ティーチング

4-19-1. ティーチング

モーションコントローラ内に記憶している運転プログラムに対してティーチング（挿入／置換／削除等）を行い、運転プログラムの変更を行うことが出来ます。

ティーチングでは記憶している運転プログラムを直接変更するので、次のプログラム実行でもティーチングした内容が反映されています。

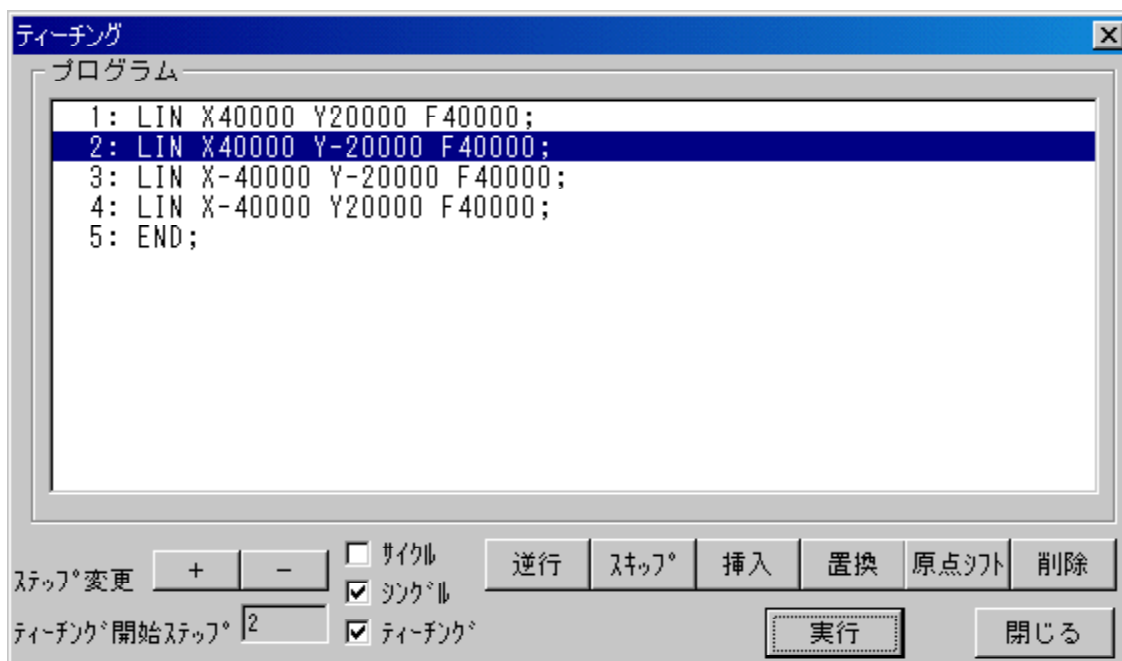
尚、ティーチングした運転プログラムはPCでアップロードして保管（管理）して下さい。

（故障時などに復旧できるようにするため）

セッティングPCのティーチング画面を元にティーチング関連の動作を説明します。

ティーチング画面で使用している通信コマンドは、全てアプリケーション開発ライブラリで公開しているので、お客様も同等の画面を作成する事が出来ます。

参考までに、以下の表にセッティングPCの操作とそれに対応する通信コマンドを併記します。



セッティングPC表示 [通信コマンド]	内 容
プログラム表示 [DAT_ONEBLOCK]	現在選択されているプログラムの内容を表示しています。 モーションコントローラ内で記憶しているプログラム情報を元に逆変換を行って表示しているため、実際にダウンロードしたプログラムとは見た目が異なっている可能性があります。
現在(強調表示)ステップ [DAT_STATUS] [DAT_TEACHSTS]	現在実行中／実行予定のステップ 又は、ティーチング対象のステップを強調表示しています。 (ティーチング中にステップ変更「+」「-」を行うと、このステップが前後します。)
ティーチング開始ステップ [DAT_TEACHSTS]	ティーチングを開始したステップを表示しています。

セッティング PC操作 [通信コマンド]	動 作	基本的条件
サイクル [REQ_CYCLE]	サイクル運転モードの設定／解除 サイクル運転モードの時は、プログラムが終了したら自動的に最初から再開します。 (ティーチング操作では使用しません。)	常に実行可
シングル [REQ_SINGLE]	シングルステップモードの設定／解除 シングルステップモードの時は、「実行」を押す毎に1ステップずつ実行します。 1ステップ実行して停止すると、「ステップ間停止」状態になっています。ほとんどのティーチング操作では「ステップ間停止」状態である必要があります。	常に実行可
ティーチング [REQ_TEACH]	ティーチングモードに設定します。 ステップ間停止中のみ可能です。 ステップ間停止にするには、シングルステップモードで実行するか、M00/M01停止を使用する必要があります。 ティーチングモードを解除します。 ティーチング移動を行っている場合は、「逆行」操作で、ティーチング開始位置まで戻る必要があります。	・ティーチングモードでない ・ステップ間停止中 ・ティーチングモード中 ・ティーチング移動未実行
ティーチング ステップ変更 ＋／－ [REQ_STEPCHG]	ティーチング対象のステップを変更します。 ステップを変更した後、ティーチング開始ステップ以外でティーチングモードを解除して再開すると、変更前と後の間のステップを実行しないで再開する事になります。注意して下さい。	・ティーチングモード中

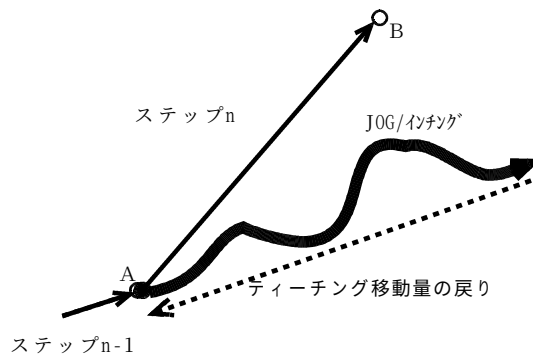
セッティング PC操作 [通信コマンド]	動 作	基本的条件
逆行 [REQ_PRGREV]	ティーチング開始位置への移動（戻り） 直前のステップの逆行。＜オプション＞	・ティーチングモード中 ・ティーチングモードでない ・ステップ間停止中 ・PTP/LINステップのみ
スキップ [REQ_STEPSKIP]	現在ステップをスキップして、次ステップに実行ステップを移します。移動中のステップは中断します。	・ティーチングモードでない ・停止中
挿入 [REQ_PRGINS]	表示されているステップ番号以降の既存ステップが繰り下げられ、ティーチング移動した分の新たなPTP指令ステップが、表示されているステップ番号として登録されます。	・ティーチングモード中 ・ティーチング開始ステップ
置換 [REQ_PRGALT]	表示されているステップの移動量が、ティーチングした移動量にさし変わります。 この時、指定コードは元のステップより継承され、アブソ移動指令であった場合は、移動先の座標値が入ることになります。また、PTMAコードの場合も同様に、指定されているポイント番号の座標データに差し変わります。	・ティーチングモード中 ・移動命令のステップ
原点シフト [REQ_ZRNSHIFT]	ティーチング移動先の論理座標がティーチング開始時の論理座標と同じになるように、論理原点をシフトします。	・ティーチングモード中
削除 [REQ_PRGDEL]	表示されているステップ番号のステップが削除され、それ以降の既存ステップが繰り上げられます。	・ティーチングモードでない ・ステップ間停止中
実行 [REQ_PROGSTRT]	プログラム運転の開始／再開を行います。	・ティーチングモードでない ・プログラム未実行／停止中

4-19-2 「逆行」動作詳細

(1) ティーチング開始位置への移動（戻り）

ティーチングで移動した分をキャンセルするPTP移動を行い、ティーチング開始位置へ戻ります。
既にティーチング開始位置にいる場合は何もしません。

- ① A点にてステップ間停止
(現在ステップ表示はn)
- ② 「ティーチング」
ティーチングモード設定
- ③ 各軸JOG、イン칭ングなどを使って移動
- ④ 「逆行」
A点にPTP動作で戻る。
- ⑤ 「ティーチング」
ティーチングモード解除

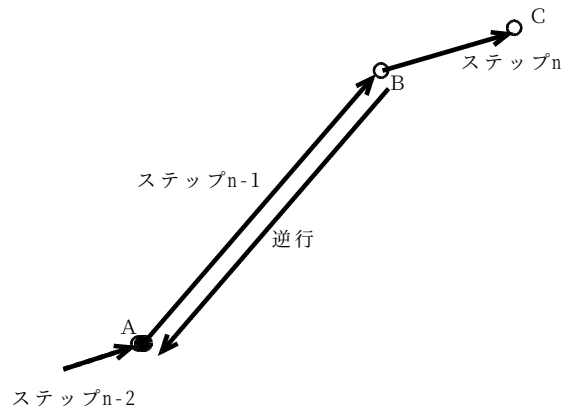


(2) 直前のステップの逆行をします。＜オプション＞

直前のステップを逆に移動して実行前の位置へ戻ります。

※ この動作はインクレ移動（PTP／LIN）のステップのみ可能です。

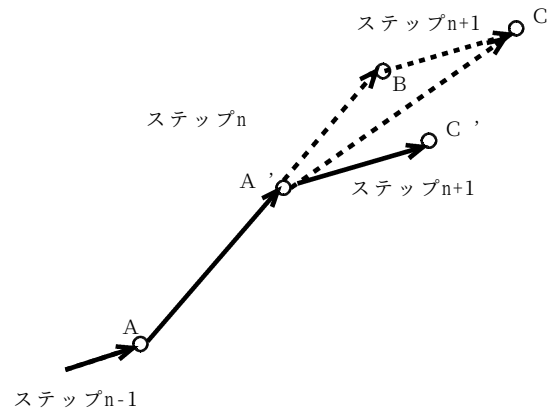
- ① B点にてステップ間停止
(現在ステップ表示はn)
- ② 「逆行」
A点へ逆移動します。
(現在ステップ表示はn-1)



4-19-3 「スキップ」動作詳細

現在ステップをスキップして、次ステップに実行を移します。移動中のステップは中断します。

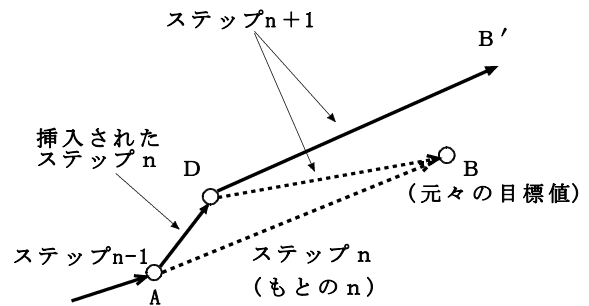
- ① A' 点にて停止
(現在ステップ表示はn)
- ② 「スキップ」
移動を中断して、次ステップに実行が移ります。
(現在ステップ表示はn+1)
- ③ 「実行」
次の移動命令がPTP/LIN(インクレ)の時は、C'へ、PTPA(アブソ)の時は、Cへ移動



4-19-4 「挿入」動作詳細

現在ステップ（ティーチング対象のステップ）以降の既存ステップが繰り下げられ、ティーチング移動した分の新たなPTP指令ステップを、現在ステップに記憶（挿入）します。

- ① A点にてステップ間停止
(現在ステップ表示はn)
- ② 「ティーチング」
ティーチングモードに入る。
- ③ 各軸JOG、オーバーライドなどを使ってD点へ移動。
- ④ 「挿入」
ステップnにD点までのPTP命令が挿入されます。
(現在ステップ表示はn+1に変わる)
- ⑤ 「ティーチング」
ティーチングモード解除
- ⑥ 「実行」
次の移動命令（もとのステップn）がPTP/LIN(インクレ)の時は、B'へ
PTPA(アブソ)の時は、Bへ移動



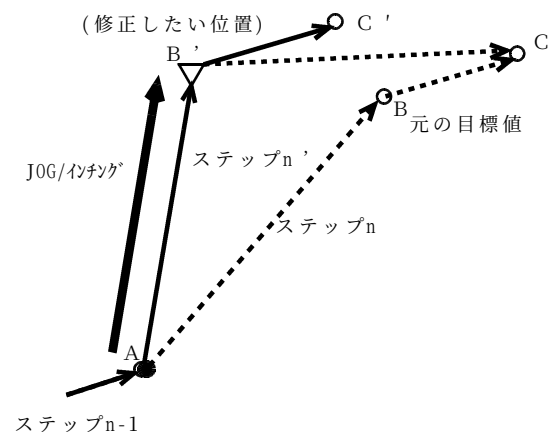
4-19-5 「置換」動作詳細

現在ステップ（ティーチング対象のステップ）の移動量が、ティーチングした移動量に差し変わります。この時、指定コードは元の命令より継承され、アブソ移動指令であった場合は、移動先の座標値が入ることになります。また、PTMAコードの場合も同様に、指定されているポイント番号の座標データが移動先の座標に差し変わります。

(1) 現在ステップ（次ステップ）の置換

※ 現在ステップが移動命令（PTP/PTPA/PTPB/LIN/LINA/LINB/PTMA）の時のみ可能な操作です。
ただし、PTMAのポイント指定をマクロ変数で指定していた場合は実行できません。又、移動量/移動座標指定がマクロ変数指定だった場合は、マクロ変数指定が解除されます。（数値指定になります）

- ① A点にてステップ間停止
(現在ステップ表示はn)
- ② 「ティーチング」
ティーチングモード設定
- ③ 各軸JOG、インチングなどを使って
B'へ移動
- ④ 「置換」
ステップnをステップn'に置換して、
現在ステップ表示がn+1に変る。
- ⑤ 「ティーチング」
ティーチングモード解除
- ⑥ 「実行」
n+1ステップから実行を再開します。
(移動命令がPTP/LIN(インクレ)の時は、
C'へ、PTPA(アブソ)の時はCへ移動)

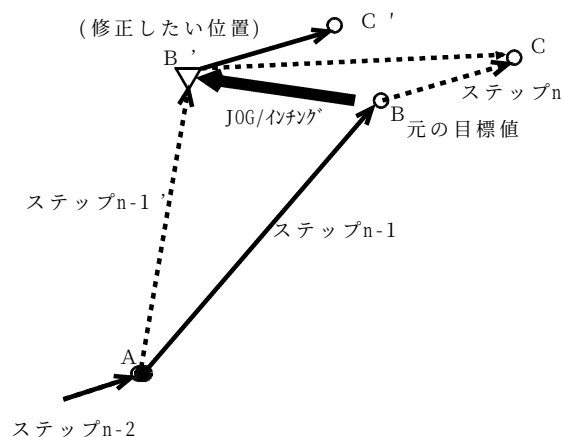


(2) 直前に実行したステップの置換（目標位置のオフセット）

※ ティーチングを開始したステップの直前のステップで、インクレ移動命令（PTP/LIN）の時のみ可能な操作です。（アブソ命令の場合は、次項の「任意のステップの置換」の処理になります。）

ただし、移動量指定にマクロ変数を指定していて、その軸を移動させた場合は、実行出来ません。

- ① B点にてステップ間停止
（現在ステップ表示はn）
- ② 「ティーチング」
ティーチングモードに入る。
- ③ ティーチングステップ「－」
現在ステップ（ティーチングステップ）を
戻します。
- ④ 各軸JOG、イン칭ングなどを使って
B'へ移動
- ⑤ 「置換」
ステップn-1をステップn-1'に置換して、
現在ステップ表示がnに変る。
- ⑥ 「ティーチング」
ティーチングモード解除
- ⑦ 「実行」
ステップnから実行を再開します。
（移動命令がPTP/LIN（インクレ）の時はC'へ、
アブソ命令の時はCへ移動）



(3) 任意のステップの置換（アブソ命令のみ）

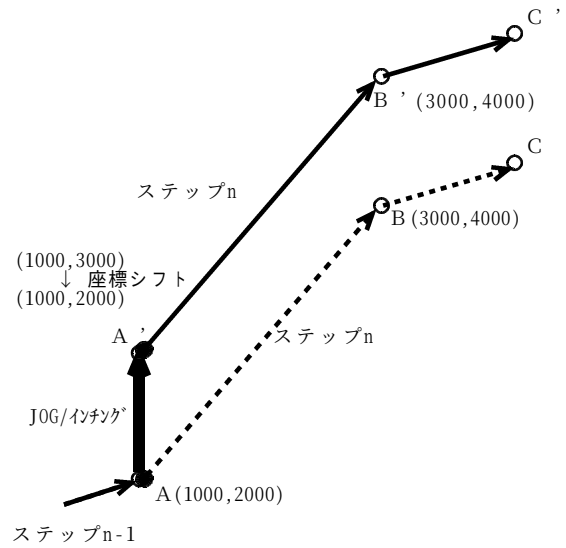
※ アブソ命令[PTPA/PTPB/LINA/LINB/PTMA]のステップに対してのみ可能な操作です。
ただし、PTMAのポイント指定をマクロ変数で指定していた場合は実行できません。又、移動座標指定がマクロ指定だった場合は、マクロ指定が解除されます。（数値指定になります）

- ① 任意のステップでステップ間停止
- ② 「ティーチング」
ティーチングモードに入る。
- ③ ティーチングステップ「+」／「－」キーにより、現在ステップを変更
- ④ 「置換」
ティーチングステップの目標座標を現在の座標に置換します。
- ⑤ 「ティーチング」
ティーチングモード解除

4-19-6 「原点シフト」動作詳細

ティーチング移動先の論理座標がティーチング開始時の論理座標と同じになるように、論理原点をシフトします。

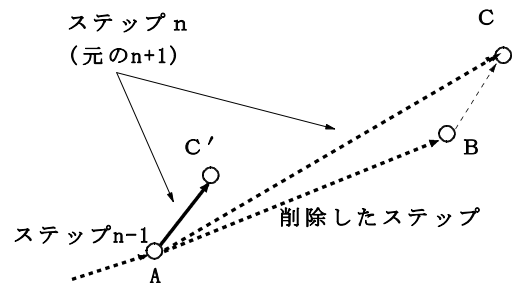
- ① A点にてステップ間停止
(現在ステップ表示はn)
- ② 「ティーチング」
ティーチングモード設定
- ③ 各軸JOG、インチングなどを使って
A'へ移動
- ④ 「座標シフト」
A'点の座標がA点の座標と同じになる
ように論理原点をシフトします。
- ⑤ 「ティーチング」
ティーチングモード解除
- ⑥ 「実行」
ステップnから実行を再開します。



4-19-7 「削除」動作詳細

表示されているステップ番号のステップが削除され、それ以降の既存ステップが繰り上げられます。

- ① A点にてステップ間停止
(現在ステップ表示はn)
- ② 「削除」
ステップnが削除されて、n+1以降の
ステップが1つずつ前にシフトしま
す。
- ③ 「実行」
次の移動命令がPTP/LIN(インクレ)の
時は、C'へPTPA(アブソ)の時は、
Cへ移動。



4-20 補間前加減速機能（自動コーナ加減速） <オプション>

4-20-1 補間前加減速のメリット

- (1) コーナ内回りの軌跡誤差の最小化
- (2) 加速度(速度変化量)を自動判別して、必要に応じて加速度(速度変化量)を制限すべく自動的に減速。
- (3) 判定や減速度合いは、SLMの自動計算ですので、運転プログラム作成者が個別に計算する必要はありません。
- (4) 判定基準や補間前加減速時定数はあらかじめ自由に設定できます。
- (5) 判定区分は6段階で、かなり自由度がありますので、機構に応じて最適な設定が可能です。

4-20-2 補間前加減速の概要

直線補間や円弧補間の連続点での加速度(速度変化量)を自動計算し、その大きさによって、適切な速度に減速/加速します。たとえば、2m/秒速度の直線から、0.5m/秒の円弧に進入するとき、直線補間の最後に減速してから、円弧に進入します。

SLMの内部処理を簡単に説明します。

軌跡発生（補間）の前に全体速度を加減速する機能を追加致します。

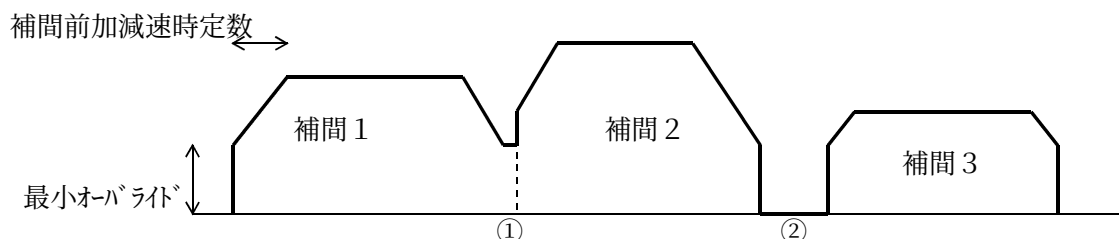
これによって、従来の（各軸）加減速の時定数を10msecのように小さくしても各軸のモータに対して、急激な加減速が発生しないようになります。



補間命令のつなぎ目で、加速度(速度変化量)をチェックし、大きいときは、その補間の減速と次の補間の加速をおこないます。

- (1) 加速度判定 補間時に、加速度(速度変化量)を計算し、その大きさを判定します。この判定を6段階に分類し、その程度に応じて、送り速度全体を減速します。
- (2) 補間前加減速 加速度判定の結果、補間のつなぎの部分で、設定された減速と加速をいたします。

4-20-3 補間前加減速をしているときの速度変化（オーバーライド）の様子



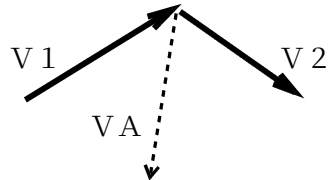
- ① 補間前加減速を必要とする補間連続点（速度変化がそれなりに大きい連続点）
オーバーライド指定なので、ステップで指令速度が違う場合は速度変動が発生します。
- ② 補間が連続しない点（ドウェルなど）

4-20-4 制限事項など

- (1) 判定値や時定数は、設計値である程度既定できますが、本来の最適値は、実際のサーボ系や機構系で再確認する必要があります。
- (2) 以下のような微小補間の場合は、減速が間に合わない場合があります。
例 ①直線補間 L1 F1 ;
②直線補間 L2 F2 ; (微小補間)
F1とF2の変化が小さいと①は減速せずに②へ移行します。この時、②の移動量が小さい微小補間の場合は、減速が間に合わず、急停止する事になります。
運転プログラムの作成時にご注意下さい。
- (3) サーボ系の位置ループ制御の影響による内回りは、発生します。ただし、これはかなり小さい値です。また、サーボのフィードフォワード制御などで、改善も期待できます。
- (4) 補間命令の間に IF, #□□□□, TIM などの補間命令以外の命令がある場合は、そのステップを速度0とみなして加速度判定します。その結果、減速/加速を行います。

4-20-5 加速度(速度変化量)の算出方法

補間命令の場合は以下の様に、つなぎ目の加速度(速度変化量)を算出します。



$$\text{加速度 } VA = V2 - V1$$

$$= \sqrt{(F2X - F1X)^2 + (F2Y - F1Y)^2 + \dots + (F2A - F1A)^2}$$

VA : 加速度(速度変化量)

V1 : 実行するステップの補間移動ベクトル
(円弧の場合は、始点の接線ベクトル)

V2 : 次ステップの補間移動のベクトル
(円弧の場合は、終点の接線ベクトル)

F1X・・・F1A : V1の各軸速度

F2X・・・F2A : V2の各軸速度

以下の組み合わせで、補間命令の加速時/減速時の速度変化量VAを算出します。

加速時：前ステップと実行ステップ

減速時：実行ステップと次ステップ

・ 前ステップが補間でない場合 : 前ステップのベクトルを0としてVAを計算します。※

・ 次ステップが補間でない場合 : 次ステップのベクトルを0としてVAを計算します。※

※補間でないステップのベクトルは0として計算するため、実行ステップの速度そのものが加速度(速度変化量)となります。

4-20-6 パラメータの設定について

セッティングPCの以下の画面で、関連のパラメータを設定します。

補間前加減速パラメータ

補間前加減速時定数
200 msec
(0: 補間前加減速無効)

最小オーバーライド設定

	加速度(速度変化量) [pls/sec]	最小オーバーライド [%]
加速度 小	0 ~ 10000 =>	100
↑	10000 ~ 50000 =>	80
	50000 ~ 100000 =>	50
	100000 ~ 150000 =>	40
	150000 ~ 200000 =>	20
↓	200000 ~ 250000 =>	10
加速度 大	250000 ~ =>	5

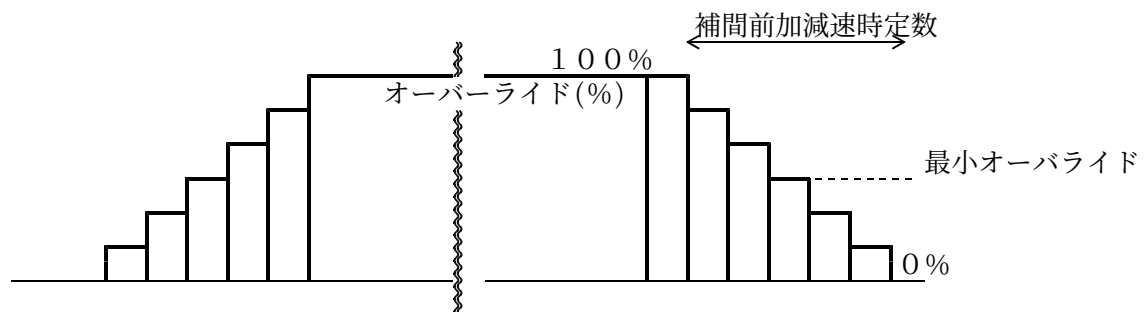
※加速度は絶対値を設定して下さい。

ファイル読出 ファイル保存 アップロード ダウンロード 編集前に戻す 閉じる

ファイル保存 設定したデータを拡張子.acoのファイルとして保存します。
ファイル読出 保存した設定データを読み出して、設定します。
アップロード SLMから現在の設定値を読み出します。
ダウンロード SLMへこの画面の設定を書き込みます。

補間前加減速時定数
(単位[msec]) 設定範囲：0 msec～1 6 3 8 3 msec(0msecを設定すると、補間前加減速機能は無効になります。)
補間前加減速オーバーライドが0%～100%に変化するまでの時間です。
(加速と減速の傾き(時間)は同じです。)
※ 傾き一定になるように変化させるので、実際の加減速時間は、その時の目標オーバーライドによって変わります。

最小オーバーライド設定 加速度(速度変化量)設定範囲：0 pps～8 0 0 0 0 0 pps
最小オーバーライド設定範囲：1%～100%
加速度(速度変化量)とそのときの最小オーバーライド(目標オーバーライド)の関係のテーブルを設定します。



減速時は、最小オーバーライド(目標オーバーライド)にかかわらず、0%まで減速できる所から減速を開始します。(最小オーバーライドまで減速した後は、最小オーバーライドの速度で動作します。)そのため、移動の終点付近では一定速になります。(減速を開始してからの時間は、補間前加減速時定数より短くなります。)

4-20-7 動作例

セッティングPCの「補間前加減速パラメータ」画面で以下の設定をした時の動作例を記述します。

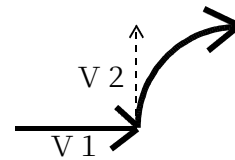
- ・補間前加減速時定数 : 330 msec
- ・最小オーバーライド

加速度(速度変化量[pls/sec])		
(表示のみ)	設定値	最小オーバーライド
0	~ 500000	(無効: 100%)
500000	~ 700000	80
700000	~ 1000000	70
1000000	~ 1200000	40
1200000	~ 1500000	20
1500000	~ 2000000	10
2000000	~	5

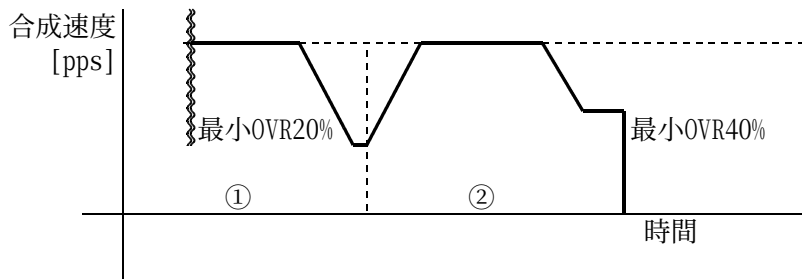
以下の<例1>~<例3>は、全て同じ速度(F指定)で直線補間→円弧補間動作を実行した例です。速度は同じですが、直線補間→円弧補間の加速度(速度変化量)が異なるため、選択される最小オーバーライドが異なります。

<例1>右図の軌跡の運転プログラム

- ①LIN X500000 F1000000;
- ②CIRR X500000 Y500000 I500000 J0 F1000000;
- ③END;

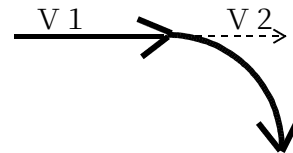


$$\begin{aligned} \text{①と②の加速度(速度変化量)} &: 1414213 = \sqrt{(0-1000000)^2 + (1000000-0)^2} \\ \text{②と③の加速度(速度変化量)} &: 1000000 = \sqrt{(1000000)^2 + (0)^2} \end{aligned}$$

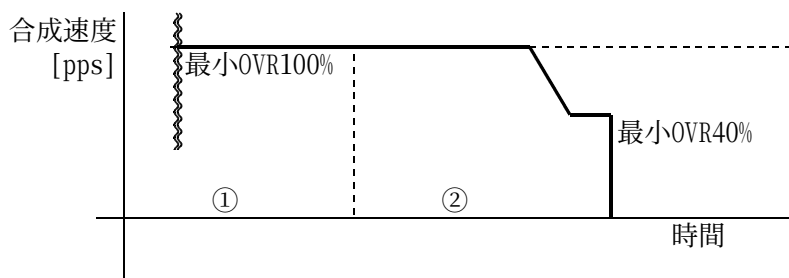


<例2>右図の軌跡の運転プログラム

- ①LIN X500000 F1000000;
- ②CIRR X500000 Y-500000 I0 J-500000 F1000000;
- ③END;

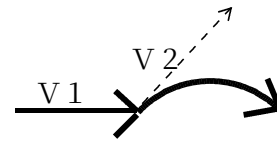


$$\begin{aligned} \text{①と②の加速度(速度変化量)} &: 0 = \sqrt{(1000000-1000000)^2 + (0-0)^2} \\ \text{②と③の加速度(速度変化量)} &: 1000000 = \sqrt{(1000000)^2 + (0)^2} \end{aligned}$$

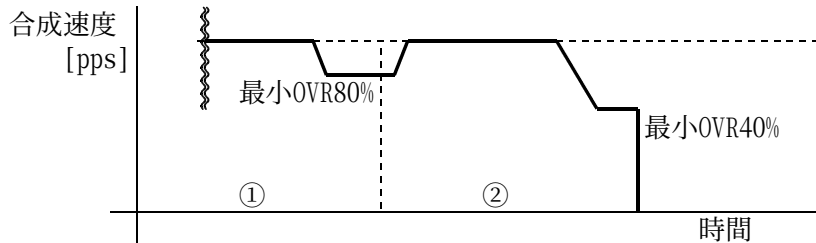


<例 3>右図の軌跡の運転プログラム

- ①LIN X500000 F1000000;
- ②CIRR PXY X707107 Y0 CR500000 F1000000;
- ③END;



$$\begin{aligned} \text{①と②の加速度(速度変化量)} &: 707107 = \sqrt{(1000000-707107)^2 + (0-707107)^2} \\ \text{②と③の加速度(速度変化量)} &: 1000000 = \sqrt{(1000000)^2 + (0)^2} \end{aligned}$$



4-20-8 パラメータの選定例

実運転時の最高速度 : 6m/分 = 100000P/秒 (1p = 1μm)
 仮に 6m/分の速度で急加速や急停止をおこなうときの加速/減速時間を0.04秒とします。

$$\text{加速度(速度変化量)} \quad 100000 \text{ pps} - 0 \text{ pps} = 100000 \text{ pps}$$

設定例

補間前加減速時定数T 40msec

最小オーバーライド

加速度(速度変化量[pls/sec])		最小オーバーライド
(表示のみ)	設定値	
0	~ 20000	(無効: 100%)
20000	~ 36000	84
36000	~ 52000	67
52000	~ 68000	50
68000	~ 84000	34
84000	~ 100000	17
100000	~	1

実際の確認

- ① 補間前加減速時定数T: 40msecの確認
 6m/分 (100000P/秒) で直角に曲がる部分で、軌跡形状を確認下さい。
 まだ、うねりがあれば、50~60msecなどにして再度トライしてください。
 逆に問題なければ、補間前加減速を30msecや20msecとして、限界を探します。
- ② その状態で、他の軌跡部を確認して、必要であれば加速度(速度変化量)の分けや最小オーバーライドを変更してください。

【注意点】

- 注1 通常の補間加減速も多少設定する事を推奨します。
 定時直線型補間加減速: 5msec~数十msec
 ただし、この数値に比例して、軌跡の内回りが発生します。
- 注2 補間前加減速を必要以上に使用すると以下の問題もあり得ます。
 ①自由曲線などのコーナ部(微小な直線補間の連続)で不自然な減速/加速が発生しがたがたした動きになってしまう。
 ②全体のタクトが下がる。減速と加速が入る分だけ、移動時間が遅くなるため。